Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

Martijn Langeweg

Service robot with force controlled walking assistance

Master's thesis in Cybernetics and Robotics
Supervisor: Jan Tommy Gravdahl
Co-supervisor: Trygve Thomessen
June 2024

NTNU
Norwegian University of
Science and Technology

Martijn Langeweg

Service robot with force controlled walking assistance

Master's thesis in Cybernetics and Robotics
Supervisor: Jan Tommy Gravdahl
Co-supervisor: Trygve Thomessen
June 2024

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Preface

This thesis is based on the groundwork established in the project report Langeweg (2023) for the Specialization project TTK4550, within the Department of Engineering Cybernetics at the Norwegian University of Science and Technology (NTNU). The ideas and insights from that work form the basis for the current study. Supervision was provided by Prof. Jan Tommy Gravdahl of NTNU and Prof. Trygve Thomessen, Managing Director of PPM Robotics AS.

This project is restricted, therefore, the code belongs to PPM Robotics AS and is not available as open source.

Acknowledgements

I would like to express my gratitude to my supervisors, Jan Tommy Gravdahl and Trygve Thomessen for their guidance and support throughout this research.

I am also grateful to PPM, specifically Bálint Tahi and Vemund Kallevik for their assistance in resolving numerous technical challenges with the robot and for Vemund's help in installing the force sensor.

Additionally, I would like to thank all the participants in my experiments for their time and cooperation.

A heartfelt thank you goes to my parents for their unwavering support, especially to my dad for offering insights from a physiotherapist's point of view.

Finally, I owe a special thanks to my girlfriend for her patience, support and for taking the time to carefully read my thesis.

Abstract

As the world's population ages, it becomes ever more critical to address the challenges of elderly care. Among these challenges is the shortage of caregivers, underscoring the importance of providing mobility support. This thesis explores the introduction of force-controlled walking assistance, utilizing the Kompaï Assist service robot for experimental and validation purposes. A central objective of this inquiry is to ensure safety and user trust, by facilitating dynamic interactions that adapt to the unique requirements of the users. The study explores how dynamic interaction, emulated by an admittance model, facilitates compliant behavior. Furthermore, haptic feedback is utilized to offer assistance and simultaneously preserve user autonomy. As such, the interaction dynamics are enhanced, while highlighting the importance of trust between the user and the robot. The study highlights the influence of different diseases and impairments on user interactions, stressing the importance of adaptability in designing robotic systems to address individual needs.

Contents

List of Tables

List of Figures

Acronyms

ADRC Active Disturbance Rejection Control. 7

AO Adaptive Oscillator. 12

CDF Cumulative Distribution Function. 27

COP Centre of Pressure. 8, 10

DMPs Dynamic Movement Primitives. 7, 10

DoF Degrees of Freedom. 9

FSM Finite State Machine. viii, 7, 10, 40, 41, 46, 47, 50

FSR Force-Sensing Resistor. 11

GA Genetic Algorithm. 9

GMM Gaussian Mixture Models. 7

GUI Graphical User Interface. viii, 5, 55, 56

HRI Human-Robot Interaction. 12, 14, 16, 18, 22, 45, 57

IBAC Intention-Based Admittance Control. 9

ICR Instantaneous Centre of Rotation. 20, 21

IMU Inertial Measurement Units. 11

ISO International Organization for Standardization. 2, 18

ITD Intentional Direction. 9, 10

LIPM Linear Inverted Pendulum Model. 8, 14

LoS Line of Sight. viii, 5, 30, 31, 52, 101

MEMS Micro-Electro-Mechanical Systems. 9

MPC Model Predictive Control. 8

MRAC Model Reference Adaptive Control. 9

NDOB Nonlinear Disturbance Observer. 8

NTNU Norwegian University of Science and Technology. i

PCI Physiological Cost Index. 12, 33, 34

PDF Probability Density Function. 26, 27

pHRI physical Human-Robot Interaction. 8

POI Point Of Interest. 86

PPF Percent-Point Function. 27

PPS Patient Performance Selector. 49

RAS Robotic Assistive System. 7, 8

RNN Recurrent Neural Network. 8

ROS Robot Operating System. 38

SPRT Sequential Probability Ratio Test. 7

STS Sit-to-Stan. 8, 15

VAC Variable Admittance Control. vi, vii, ix, 7, 32, 53, 55, 76, 79, 81–83, 97, 99, 100

WHO World Health Organization. 10

WMM Wheeled Mobile Manipulators. 7, 10, 15

XTE Cross-Track Error. 35, 83, 85

1

Introduction

The challenges posed by the aging population and the scarcity of healthcare professionals present profound implications for ensuring the quality of elderly care in the coming years. Projections indicate a shortage of 40,000 nurses for elderly care by 2035, and by 2060, Norway's workforce will be insufficient to meet the needs of both the industrial and elderly care sectors (*Meld. St. 7 (2019–2020) - regjeringen.no* (2023)). Consequently, there arises a critical necessity for technological interventions to effectively address and mitigate these impending challenges (Langeweg (2023)).

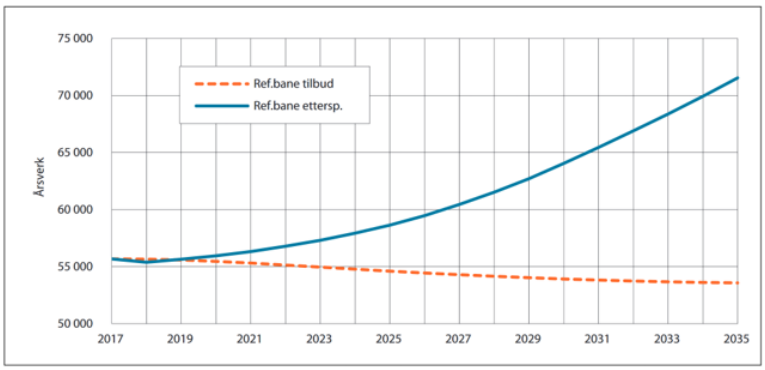


Figure 1.1: Predicted need for healthcare workers in elderly care (*Meld. St. 7 (2019–2020) - regjeringen.no* (2023)).

1.1 Background

This thesis expands upon the groundwork laid out in the specialization project (Langeweg (2023)). As such, the initial paragraphs of the Background (Section 1.1) and Problem Description (Section 1.2) are based on the project report.

Mobility aids have become increasingly important in addressing the needs of the expanding elderly population and mitigating the shortage of caregivers. For instance, canes and walkers are commonly utilized to enhance balance and mobility. They play a vital role in providing stability, supporting the user's weight, and facilitating movement (Naeem et al. (2022)). Nevertheless, traditional designs of both canes and walkers suffer from certain limitations. They often lack maneuverability and demand a significant amount of propulsion force (Xing et al. (2021)). Consequently, this presents a challenge for seniors with weakened lower limb muscles. These challenges underscore the necessity for design and technological enhancements to optimize the effectiveness of walking aids (Langeweg (2023)).

Considering these challenges, this study investigates novel literature concerning intelligent and adaptable service robots to provide mobility support for elderly individuals. The International Organization for Standardization (ISO) defines a service robot as a "robot in personal use or professional use that performs useful tasks for humans or equipment" (*ISO 8373:2021, Robotics* (2023)). These robots require a certain level of autonomy, which is the "ability to perform intended tasks based on current state and sensing, without human intervention" (*ISO 8373:2021, Robotics* (2023)). By performing tasks without constant human intervention, service robots address the caregiver shortage and provide more stable walking assistance (Langeweg (2023)).

In addition to addressing the challenges posed by an aging population and the shortage of healthcare professionals, it is crucial to recognize the significance of fall prevention among elderly. Falls represent a significant risk for older adults, where rollover in the sideways direction is identified as the most frequent type of fall (Bilgin et al. (2023)). The consequences of such falls may be severe, often leading to hip fractures due to inappropriate timing during balance recovery (Ilic et al. (2023)). Hip fractures not only impose a considerable burden on healthcare systems but also result in increased mortality rates and reduced functional abilities, ultimately compromising the quality of life for affected individuals (Ilic et al. (2023)).

Technologies such as cane-type robots (Cerqueira et al. (2023)) and smart walkers equipped with fall-prevention strategies (Pereira et al. (2019)) have emerged as viable options to mitigate the risk of falls and enhance the safety of elderly individuals. The implications of fall prevention extend beyond physical health outcomes. By maintaining mobility and independence, fall prevention strategies contribute to improving quality of life and prolonging the ability for older adults to remain at home (Mahdi et al. (2022)). This aligns with the preferences of many elderly individuals who wish to stay in familiar environments to preserve autonomy and privacy (Zhao et al. (2020)).

Moreover, non-wearable devices, exemplified by cane robots and smart walkers, offer walking support without constraining the user's leg movement (Itadera, Nakanishi, et al. (2020)). These devices enable natural gait patterns and allow for the evaluation of walking abilities in situations closely resembling real-life scenarios. On the contrary, wearable devices such as exoskeletons or exosuits directly augment the user's movements, offering targeted support tailored to individual needs. However, they often require intricate attachment processes and may restrict natural motion (Itadera, Nakanishi, et al. (2020)). Consequently, wearable devices will not be considered within the scope of this study.

To address the aforementioned challenges, this study utilizes the Kompaï Assist robot,

a novel service robot developed by Kompaï Robotics (*KOMPAI-Assist | Kompai Robotics* (2023)). It serves as an experimental platform to investigate the use of intelligent mobility devices in healthcare. The Kompaï robot offers physical and cognitive support, aiding patients in reaching their destinations. Beyond serving as a mobility aid, Kompaï plays a crucial role in health monitoring and providing entertainment, thereby enhancing the overall quality of care. Additionally, Kompaï functions as a logistics partner, capable of transporting goods (*KOMPAI-Assist | Kompai Robotics* (2023)). Consequently, it allows caregivers to devote more time to patients, which improves the accessibility and availability of care (Langeweg (2023)).

1.2 Problem Description

Expanding on the defined capabilities of the Kompaï robot, this study aims to investigate and evaluate various force control strategies tailored specifically for elderly mobility. The primary objective is to provide compliance between the user and the robot by offering guidance and walking support (Langeweg (2023)). Additionally, the study aims to explore fall prevention measures within the walking assistance framework. Ultimately, the utilization of the robot is intended to assist the elderly in reaching their destinations, while concurrently enhancing their physical activity and independence.

Guiding entails the service robot actively assisting and directing individuals in their movements from one location to another. The robot guides the users along a predetermined route, providing precise and secure navigation from point A to point B. Consequently, it ensures user safety and confidence during movement. Notably, users can command the robot to initiate motion through interactive forces. Hence, the robot must respond to user-applied forces, allowing it to adapt to the user's intentions while remaining in proximity to the predetermined trajectory. This method offers users a degree of freedom while maintaining proximity to the trajectory, striking a balance between support and user autonomy (Jiménez et al. (2019)).

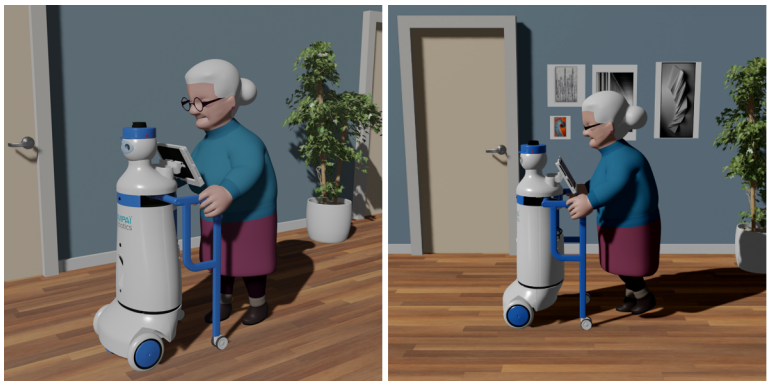


Figure 1.2: Walking support with the Kompaï robot (*PPM Robotics AS* (2023)).

In the context of walking support, the service robot offers assistive forces as users com-

mand motions. Users possess the freedom to direct the robot, allowing for controlled motions guided by their intention. The robot provides assistive forces to ease control for users with weakened muscles, which is especially beneficial for elderly individuals (Pereira et al. (2019)). Additionally, the robot actively introduces stabilizing motions to prevent falls during movement. Consequently, the walking support enhances stabilization and maneuverability (Naeem et al. (2022)). Beyond basic assistance, the robot incorporates training elements by integrating information, motivation, and challenges throughout the motion. This approach aims to improve the user's walking experience, prioritizing safety and engagement through personalized and supportive interactions (Langeweg (2023)).

Lastly, fall prevention measures focus on helping users recover their balance after a loss of stability. Yearly, more than a fourth of individuals over 65 years old fall (*Falls and Fractures in Older Adults* (2022)). Many older adults fear falling, which can lead them to avoid activities like walking. However, maintaining an active lifestyle can reduce muscle weakness and help prevent falls. For elderly individuals, a fall can have severe consequences, including loss of independence, chronic pain, and reduced quality of life (Vaishya et al. (2020)). Walking aids, such as robotic walkers, provide stability and lower the risk of falling. Given the potential severity of falls, intelligent walker aids should include features that help users regain balance during emergencies. Thus, implementing fall prevention measures can significantly enhance user safety and quality of life.

1.3 List of Contributions

In this section, the significant contributions of the thesis are outlined. These contributions represent advancements in mobility support for the Kompaï robot. Methods for walking support, fall prevention, and guidance are based on a literature review and tailored to the specific capabilities of the robot. The main contributions are summarized as follows:

- **Turning strategies:** Developed turning strategies for a differential drive robot to adjust the turning radius by the applied force.
- **Force sensor:** Incorporated a force sensor on the Kompaï robot and configured it with the appropriate transformation matrix.
- **Reconstruction of handle forces:** Reconstructed handle forces from measured forces in the sensor frame to assess user intention.
- **Admittance controller for walking support:** Implemented an admittance controller proposed by Itadera, Nakanishi, et al. (2020) for a cane robot and adapted it for walking support with the differential drive Kompaï robot.
- **State machine for system state:** Incorporated a state machine governing the solution's state transitions, determining when to initiate the developed controller and how to interpret human actions, such as user presence or stopping due to imbalance.
- **Stability assessment:** Developed a relationship between timestep, mass, and damper parameters in the admittance controller to assess stability.

- **Fall prevention:** Implemented a fall prevention strategy by monitoring the user's walking state within a support ellipse and adjusting admittance parameters accordingly, as suggested by Hirata et al. (2006).
- **Confidence ellipse estimation:** Utilized the Mahalanobis distance and its Chi-Square distribution to estimate a 90% confidence ellipse of the user's position, thereby enabling the fall prevention strategy when the user moves outside this ellipse.
- **Path planning:** Exploited the Line of Sight controller (LoS) developed by Fossen et al. (2003) to generate a steering angle to navigate a set of waypoints, thereby enabling the implementation of the guidance.
- **Guidance through haptic sensations:** Implemented guidance with the Kompaï robot, utilizing haptic sensations inspired by Jiménez et al. (2019), and selected appropriate coefficients for the function governing adjustments in admittance parameters.
- **Guidance through Kompaï's planner:** Implemented a second guidance solution using the Kompaï robot's path planner and an admittance controller in the forward direction to regulate linear velocity.
- **Performance metrics:** Developed performance metrics to evaluate the implemented solution during experiments with human subjects.
- **Graphical User Interface:** Implemented a Graphical User Interface (GUI) with FlexGui 4.0 (*FlexGui 4.0* (2022)) to facilitate easier startup and recording of performance metrics during experiments.

1.4 Report structure

The report is composed of nine chapters. Chapter 1 establishes the study's background and defines the identified problem. Chapter 2 conducts a literature review on innovative force control solutions, while Chapter 3 addresses trust in automation.

Moving forward, Chapter 4 delves into the theoretical framework encompassing human-robot interaction. Chapter 5 details hardware and software solutions specific to the Kompaï robot, followed by the methodology in Chapter 6.

Moreover, Chapter 7 presents experimental results, while Chapter 8 discuss findings, limitations and future considerations. The report concludes in Chapter 9 where the key findings are summarized while emphasizing the significance of the research.

2

Literature review

To establish the framework for this study, the literature review builds upon the groundwork established in the preceding project report by Langeweg (2023). The preliminary investigation yielded valuable insights into force-controlled walking assistance. While Langeweg (2023) contributed to the initial understanding, the present literature review expands its focus to explore strategies related to fall prevention, guidance and dynamics of the human operator. Accordingly, Section 2.2 and 2.3, along with the review addressing some articles in Section 2.1, 2.4, 2.5 and 2.8, are extracted from Langeweg (2023).

An area that is gaining attention, is the use of force control in smart walkers designed for elderly care. As the global population ages, there is a growing demand for solutions that support elders in maintaining their independence and well-being (Langeweg (2023)). The shortage of nursing professionals and the increasing number of elderly people has prompted the development of mobility assistive devices, rehabilitation robots, and nursing care robots as a solution (Cerqueira et al. (2023)).

In order to assess the need for assistive walkers in elderly care, a search in the Scopus database was conducted on the 11th of March 2024. Furthermore, reproducibility is made possible via a Scopus literature search, enabling a study that is representative of the state of the art. The primary search criteria (Appendix A) included elderly using the logical operators AND and OR to efficiently narrow down the results. These terms were combined to cover topics including assistive walking devices, smart/robotic walkers, mobility, force control, admittance control, impedance control, compliance, and robot. The exclusion of specific terms like exoskeleton, exosuit, wheelchair, prosthesis, humanoid and wearable made it possible to sort out irrelevant results. Furthermore, the date range for this search spanned from 2018 to 2023, to guarantee the inclusion of the most recent research and technological advancements in the field of aged care and assistive walkers (Langeweg (2023)). Moreover, for the extended literature review, fall prevention is added to the search query.

2.1 Smart Walkers

Smart walkers are frequently addressed in the existing literature. Zhao et al. (2020), Chang et al. (2021) and D. Ding et al. (2023) explore the application of force sensors in smart walkers with the objective of estimating gait or human intention to eventually lower the risk of falls among elderly users.

In Zhao et al. (2020) the Smart Robotic Walker is an elderly mobility assistance device that combines a traditional rollator design with a mobile robotic platform. With its soft sensing handle, which can perceive gait posture and pressure, the walker can adapt its position in response to the user's intentions. Moreover, with its Finite State Machine (FSM) it possesses the capacity to detect emergencies, such as falls, by constantly monitoring handle pressure.

The smart walker in Chang et al. (2021) features a posture assessment that utilizes fuzzy rules to detect six distinct postures based on grip forces, walker speed, and road incline. Based on the detected posture, the walker offers supportive forces for smoother and safer mobility. In addition, users have the flexibility to customize the level of assistance to suit their requirements. Moreover, the walker prevents collisions and maintains postural stability, thus reducing the risk of potential falls.

The ReRobo walker explored in D. Ding et al. (2023) is tailored to mitigate the heightened risk of falls among the elderly. It utilizes an FSM with ready, normal, and abnormal states. In the normal state, Active Disturbance Rejection Control (ADRC) ensures responsive steering and forward movement based on user force. Additionally, upper limb forces from the force sensor and lower limb movement from the laser range finder are fused by a Kalman filter. Falls are detected via a Sequential Probability Ratio Test (SPRT) on the Kalman filter's predicted value, prompting a transition to the abnormal state upon detecting danger. This transition triggers protective actions, highlighting the device's approach to fall prevention.

2.2 Wheeled Mobile Manipulator

An alternative approach for offering assistive devices to the elderly involves the utilization of Wheeled Mobile Manipulators (WMM). This approach is examined by L. Ding et al. (2022) and Xing et al. (2021).

L. Ding et al. (2022) present a WMM to assist elderly individuals in their everyday tasks. In terms of mobility assistance, it utilizes Dynamic Movement Primitives (DMPs) to learn the user's walking patterns and provide assistive forces. Additionally, by employing Variable Admittance Control (VAC), the system is designed to detect and react to user intentions. The system also addresses unexpected situations to avoid user falls.

Furthermore, for tasks involving the manipulation of heavy objects, L. Ding et al. (2022) employs stiffness estimation and Gaussian Mixture Models (GMM) to reproduce user actions in the vertical dimension and achieve a compliant movement in the horizontal plane.

The WMM in Xing et al. (2021) integrates Cartesian-space admittance control with null-space control to improve its Robotic Assistive System (RAS) performance. This strategy offers compliance in the horizontal plane while preserving stiffness in the vertical

plane to ensure reliable support. By applying null-space control, the system maximizes its ability to exert force, enabling it to provide increased vertical support. Furthermore, the RAS utilizes a Nonlinear Disturbance Observer (NDOB) which eliminates the need for a costly force/torque sensor.

2.3 Optimization of Admittance Parameters

The concept of an admittance controller is common in assistive devices, allowing for a compliant behavior. Real-time adjustments of admittance parameters enable optimal support according to the current conditions. This approach is implemented in Itadera and Cheng (2022) and Itadera, Dean-Leon, et al. (2019), in which a human gait model is employed for the optimization of admittance parameters.

Itadera and Cheng (2022) present an algorithm to enhance the gait stability of elderly individuals by optimizing admittance control parameters. It utilizes a Linear Inverted Pendulum Model (LIPM) to represent human gait. With a Model Predictive Control (MPC) framework, the admittance controller is optimized, for the generation of a virtual assistive force to enhance the balance of human gait.

Itadera, Dean-Leon, et al. (2019) present a non-wearable robotic system designed to assist elderly individuals in walking. To improve the physical Human-Robot Interaction (pHRI), the robotic system employs an admittance controller and integrates an MPC framework for adaptive assistance. Furthermore, by utilizing a LIPM for human gait, the optimization process generates an optimal assistive force that adjusts the walker's Center of Pressure (COP), enhancing the user's balance while preserving natural gait.

2.4 Sit-to-Stand Transitions

Inadequate strength during Sit-to-Stand (STS) transitions is a major cause of falls among the elderly. Mahdi et al. (2022) and Itadera, Kobayashi, et al. (2019) investigate methods to assist with STS transitions, aiming to enhance the independence of elderly individuals (Langeweg (2023)).

SkyWalker presented in Mahdi et al. (2022) is a rollator designed to provide aid during STS transitions and walking across diverse types of terrain. The STS assistance involves applying vertical force and forward translation. The control parameters are obtained through experiments examining the kinematics and kinetics of individuals utilizing the SkyWalker.

On the other hand, the approach in Itadera, Kobayashi, et al. (2019) utilizes a mobile robot to offer assistive support for elderly individuals, particularly during STS transitions and walking. Moreover, an impedance controller is employed in conjunction with a Recurrent Neural Network (RNN) to classify the user's movement states across six distinct classes. The current user state is used to dynamically adjust the impedance parameters to provide appropriate physical support.

Another alternative is presented in Yokota et al. (2019) and aims to address the needs of elderly and disabled individuals who require assistance during STS transitions, while allowing them to use their remaining physical strength. The technology combines damping

control and position control to provide support while also permitting users to utilize their physical strength. Damping control enables the adjustment of supportive forces, altering the user's experienced load. This dynamic control mechanism offers flexibility by allowing deviations from a fixed reference pathway, ensuring that users can move freely during the STS transition (Yokota et al. (2019)). Additionally, the algorithm incorporates position control to maintain body posture when the user's posture exceeds acceptable limits. In emphasizing a user-centric approach, the robot adjusts to variations in posture tolerance, allowing users to maintain muscle exercise and autonomy.

2.5 Cane Robot

Using cane robots is an additional approach to address the growing demand for assistive devices among older individuals.

Naeem et al. (2022) present a four Degrees of Freedom (DoF) cane robot for fall prevention and improved walking support for the elderly. Compared to conventional walkers, this cane robot offers greater maneuverability. It operates in two modes: normal walking, utilizing kinematic control to follow the user, and fall prevention, employing real-time impedance control to respond to potential falls. The technology employs Model Reference Adaptive Control (MRAC) and Genetic Algorithm (GA) gains tuning to handle system uncertainties, enabling the cane robot to successfully adjust to user needs.

B et al. (2023) introduces an assistive walking cane robot equipped with force sensors in the handlebar, an ultrasonic sensor for obstacle detection, and Micro-Electro-Mechanical Systems (MEMS) sensors to identify falls. This technology ensures precise control of the cane's movement based on user exerted force and alerts through a speaker when obstacles are detected. The integration of a MEMS sensor to measure acceleration and the potential use of EEG sensors enhances the system's ability to detect emergencies.

Furthermore, the future application of machine learning algorithms aims to provide predictive analysis for advanced fall prevention. In essence, this approach aims to provide timely alerts and notifications, ultimately reducing the risk of accidents and enhancing the overall safety of the users (B et al. (2023)).

The smart cane-type robot presented in Itadera, Nakanishi, et al. (2020) employs Intention-Based Admittance Control (IBAC). This controller utilizes the concept of Intentional Direction (ITD) in modeling human walking intention. The IBAC scheme improves walking comfort by introducing two distinct admittance models, one for motion along the intended direction and another for motion perpendicular to it. These models have mass and damping parameters, enabling the cane to be easily maneuvered in the ITD while offering resistance in the perpendicular direction. The admittance control not only enhances the robot's ability to follow user intention but also contributes to effective gait training.

2.6 Estimating Human Intention

The majority of the explored literature emphasizes the importance of estimating human intention. By understanding and adapting to human intention, the aids effectively serve their users' needs while providing enhanced mobility and stability.

For Smart Walkers or WMMs, the most common approach involves employing a force sensor in conjunction with a FSM (Zhao et al. (2020); D. Ding et al. (2023)) or fuzzy rules (Chang et al. (2021)) to detect user states and emergency events. However, more sophisticated approaches leverage machine learning techniques such as neural networks (Zhao et al. (2020)) or DMPs (L. Ding et al. (2022)) to learn human intention from gait data or classify the user's state across distinct classes (Itadera, Kobayashi, et al. (2019)). Once the user's intention is determined, these methods utilize the force data as the control input to the admittance controller. Hence, the walking aid is steered toward the direction aligned with the user's intention.

On the contrary, cane robots, which provide less stability, require advanced methods to infer human intention. In this context, Cerqueira et al. (2023) classifies human intention into five distinct motions. Furthermore, Wakita et al. (2013) present a Kalman filter approach for estimating the ITD, which is employed by Itadera, Nakanishi, et al. (2020). The Kalman filter utilizes measured forces and distinct walking mode transitions. By integrating these inputs into the state model, the Kalman filter estimates the ITD.

Given that walker robots adhere to nonholonomic constraints, they benefit from increased stability and decreased sensitivity to force variations (Batlle et al. (2009)). Consequently, the estimation of ITD is not as critical for walker robots as for cane robots. Cane robots encounter challenges such as zigzag trajectories due to stumbling, making an accurate estimation of ITD essential for designing an effective motion controller (Wakita et al. (2013)).

2.7 Fall prevention

According to the World Health Organization (WHO), falls are the second leading cause of accidental or unintentional injury deaths worldwide (*Falls - WHO* (2024)). There are several risk factors that increase the probability of falling, such as poor balance and lower extremity weakness (Pereira et al. (2019)). To address these challenges patients with balance impairment often resort to using walkers. Recognizing the global impact of falls as a significant cause of unintentional injuries, the current literature actively explores avenues for fall prevention to mitigate these risks.

Fall prevention algorithms employ a variety of techniques, such as the analysis of movement patterns through gait or COP estimation to identify potential loss of balance (Itadera and Cheng (2022); Naeem et al. (2022)). Additionally, utilizing machine learning techniques is an effective method for gait analysis (Itadera, Kobayashi, et al. (2019)), enabling the distinction between normal force variations and those that indicate instability. Real-time processing must be prioritized in the design in order to ensure timely interventions and increased effectiveness of the algorithm.

Itadera, Nakanishi, et al. (2020), which was presented in Section 2.5, incorporates a safety mechanism with fall prevention capabilities. This involves monitoring the risk of a user falling based on the size of the support polygon formed by the robot base and the user's foot positions. When the risk of falling increases, the admittance parameters are dynamically adjusted to provide appropriate physical support, enhancing stability and reducing the likelihood of a fall.

Furthermore, Itadera, Nakanishi, et al. (2020) outline a virtual friction model within

its control system to improve user support and prevent falls. In traditional admittance control, the cane robot's velocity remains non-zero as long as external force is applied. Consequently, when the user leans on the cane robot for support, it continues to move and cannot adequately assist. To address this, a virtual friction force is introduced into the admittance control method. This approach facilitates the cane robot's smooth movement when the user intends to navigate it, while ensuring a prompt halt to provide support when the user leans against the robot. The virtual friction force dynamically adapts based on the vertical component of the user's applied force, delivering supportive forces to the user.

The cane-type robot presented by Cerqueira et al. (2023), behaves in accordance with the user's intentions and meets safety conditions by employing a haptic sensing system. The user's intent is detected through the robot's axial force system, which is composed of two parts: a haptic sensing system and an axial force system. The haptic sensing system, measured through vertical forces, monitors the user's gait and quantifies the body weight support. It facilitates fall detection by identifying sudden changes in the user's gait or movement patterns, offering real-time feedback to detect when support is required.

Moreover, the axial force system, comprising four Force-Sensing Resistors (FSRs), categorizes user motion intentions as front, back, left, and right, with stop indicating no applied force. Combined with a vibratory actuation mechanism employing vibrotactile motors, this facilitates two-way communication to provide feedback on hazards or gait irregularities (Cerqueira et al. (2023)).

Bilgin et al. (2023) employs lateral support mechanisms on a smart walker to reduce the risk of sideways rollovers and improve fall prevention, especially for the elderly and those with Parkinson's disease. It utilizes a combination of a laser range finder, a 6-axis force/torque sensor, and an Inertial Measurement Unit (IMU) to estimate the human state. The system responds to signs of instability, triggered either by exceeding critical force/torque thresholds or an extended swing phase during gait. This approach differs from traditional fall prevention methods. Instead of merely stopping the walker, it repositions support legs to generate a reactive force to counteract tipping torque. With a specific focus on addressing sideways falls, which are frequently observed among older adults (Bilgin et al. (2023)), this technology aims to significantly enhance user safety, stability, and balance during mobility.

2.8 Training

As individuals age, they experience a natural decline in muscle strength (Abdalla et al. (2021)). Furthermore, the strength of lower limb muscles significantly correlates with balance, risk of falling, and walking ability (Li, Y. Yamada, K. Yamada, et al. (2022)). Consequently, certain solutions go beyond traditional assistive devices by focusing on training programs tailored to improve the balance of older individuals.

In Li, Y. Yamada, Wan, et al. (2019) a gait-phase-dependent control strategy emphasizes increasing muscle power rather than just strength in order to restore balance. With a six-axis force/torque sensor, the smart walker anticipates user intentions and offers real-time estimations of gait parameters. By dynamically adjusting resistance throughout the gait, the walker has the potential to significantly improve lower limb muscle activation (Li, Y. Yamada, Wan, et al. (2019)).

Li, Y. Yamada, K. Yamada, et al. (2022) explores gait phase-dependent control, taking into account that different muscles contribute to braking force in the early stance phase and propulsion force in the late stance phase. Traditional approaches apply constant external resistance, which may neglect muscle training during braking. In contrast, Li, Y. Yamada, K. Yamada, et al. (2022) exerts resistance only when the targeted muscles are activated and releases resistance when the muscles are at rest. This ensures effective training, without interfering with natural gait patterns or causing unintended propulsion during braking phases. In conclusion, Li, Y. Yamada, K. Yamada, et al. (2022) demonstrate that effective strength training for enhanced walking performance focuses on muscles responsible for generating propulsion during walking.

The control strategy in Li, Y. Yamada, K. Yamada, et al. (2022) requires real-time estimation of the gait phase, which is accomplished using a six-axis force sensor. The walker's handle supports the user's body weight, alleviating the lower limbs by transferring force to the upper limbs. Moreover, the fluctuation of the user's center of mass during walking causes a cyclic change in the interaction force with each gait cycle. Consequently, Adaptive Oscillators (AOs) are employed to extract frequency and phase information from the oscillatory gait component, which is utilized to derive the necessary gait parameters.

Furthermore, physiological cost evaluation is a key aspect addressed by Itadera, Nakanishi, et al. (2020), presented in Section 2.5, to optimize the admittance control parameters for gait training. The evaluation involves a Physiological Cost Index (PCI), which is defined as a measure of energy consumption during walking with the cane robot. In addition, Itadera, Nakanishi, et al. (2020) considers the total work done by the user on the robot and the impulse transmitted from the user to the robot. By evaluating these physiological indicators, the study aims to assess the walking load during gait training. This emphasis on physiological cost evaluation contributes to the overall effectiveness of the cane robot in facilitating rehabilitation and enhancing the mobility of elderly and impaired individuals.

2.9 Human operator dynamics

Comprehending the dynamics of human operators in Human-Robot Interaction (HRI) is crucial for optimizing system performance. As explored by Woollacott et al. (1988), age-related neuromuscular changes significantly impact motor coordination, leading to delays in postural muscle responses and loss of coordination between muscle groups. These findings underscore the importance of acknowledging and addressing age-related factors in HRI interfaces to enhance the overall user experience. Moreover, modeling human behavior provides insight into how human operators adjust their actions to align with system dynamics. Within manual control systems, methodologies such as the crossover model (D.T. McRuer et al. (1967)) and time-optimal control theory (D. McRuer (1980)) are commonly employed to describe tracking errors.

Woollacott et al. (1988) presents the effects of neuromuscular changes on motor coordination. The research identifies significant delays in postural muscle responses among older adults when confronted with external balance threats, potentially disrupting timing and coordination. Specifically, elderly induviduals exhibited a significant increase of 73 ms in postural response latencies for the flexor muscle tibialis anterior (Woollacott

et al. (1988)). Furthermore, observed breakdowns in muscle response correlation indicate a loss of coordination between muscle groups, impacting the smoothness and efficiency of movements. Moreover, the increased co-contraction of agonist and antagonist muscles suggests a weakened ability for selective muscle activation, potentially leading to higher stiffness. These findings emphasize the importance of recognizing age-related neuromuscular changes and their implications for refining man-machine interfaces to better accommodate the needs and capabilities of elderly operators.

McRuer's Crossover Model states that the human controller (Y_p) adapts its behavior to match the system dynamics (Y_c), ensuring that the open-loop system (Equation 2.1) behaves as a single integrator with effective time delay τ_e in vicinity of the crossover frequency ω_c (Bachelder et al. (2019)). The model acknowledges the human controller's capacity to generate lead (T_L) or lag (T_I) responses using prediction or memory. Moreover, by adapting the gain K_p and reducing effective time delay τ_e the operator (Equation 2.2) achieves the desired crossover frequency ω_c (D.T. McRuer et al. (1967)). This frequency, representing the system bandwidth, signifies the upper limit where operator input effectively minimizes tracking errors (Bachelder et al. (2019)). Once adapted to the dynamics, the operator can find the optimal parameters to mitigate disturbances and eliminate tracking errors.

$$Y_{OL}(j\omega) = Y_p(j\omega)Y_c(j\omega) = \frac{\omega_c}{j\omega}e^{-j\omega\tau_e} \tag{2.1}$$

$$Y_p = K_p\frac{(T_L j\omega + 1)}{(T_I j\omega + 1)}e^{-j\omega\tau_e} \tag{2.2}$$

Time-optimal control theory presents an alternative perspective on human operator dynamics within manual control systems, focusing on minimizing the time needed to achieve specific control objectives. This framework explores how individuals adjust their actions to meet performance criteria and effectively complete tasks (D. McRuer (1980)).

Furthermore, by highlighting the importance of speed and efficiency, this theory reveals the strategies and decision-making processes operators employ to enhance performance. These decision-making processes involve trade-offs between speed, accuracy, and effort, with operators adapting and learning from experience to refine their approaches for rapid and precise task completion (D. McRuer (1980)). In essence, time-optimal control theory offers valuable insights into time-sensitive aspects of human behavior in manual control systems.

The dynamics of a human operator, particularly in terms of response delay, have been extensively studied over the past century (Bates (1947)). In basic tests, skilled operators typically exhibit a mean response time of 0.15 seconds, while novices show longer mean response times ranging from 0.2 to 0.25 seconds (Bates (1947)). An example of a test to capture this delay involves responding to a visual stimulus by pressing a button. Trained operators consistently outperform novices, showcasing the significance of experience and practice in reducing response delays and improving overall performance (Bates (1947)).

In summary, understanding human operator dynamics is essential for optimizing system performance and user experience. Age-related neuromuscular changes necessitate the adaptation of interfaces to accommodate older users' needs (Woollacott et al. (1988)). Models to describe human behavior emphasize the importance of adaptive behavior,

response time and experience in enhancing efficiency. Acknowledging these factors is critical for designing effective systems that cater to diverse user capabilities.

2.10 Summary

The comparison of the gathered articles on assistive technology for the elderly is shown in Table 2.1. These publications collectively emphasize the importance of stability and safety in supporting elderly individuals. The literature introduces control algorithms that recognize and react to human intentions, resulting in a smoother and more reliable walking experience. Such control methods prioritize stability, security and user customization in order to improve mobility support (Langeweg (2023)).

By utilizing HRI, intelligent walkers are able to recognize and react to user intentions, which enhances the overall stability and confidence of elderly users. As such, assistance for seniors is customized to meet their unique needs and preferences. This individualized approach improves the smart walker's usefulness and efficiency. In addition, the research includes gait models, such as the LIPM. Through the integration of these models, assistive forces can be optimized, contributing to mobility while preserving a smooth and comfortable gait. Overall, the studies stress the need for resolving emergency scenarios, such as falls (Langeweg (2023)).

Falls pose a significant global health risk (*Falls - WHO* (2024)), particularly for individuals with poor balance and lower extremity weakness, leading to increased reliance on assistive devices such as walkers (Pereira et al. (2019)). Fall prevention strategies leverage sophisticated algorithms and machine learning techniques for real-time analysis of movement patterns. The objective of these methods is to improve stability and minimize the risk of falls in walker robots, ultimately enhancing user safety during walking.

Understanding human operator dynamics in HRI is essential for optimizing system performance. Age-related neuromuscular changes underscore the necessity of addressing age-related factors in HRI interfaces to enhance user experience (Woollacott et al. (1988)). Modeling human behavior provides insights into how operators adapt their actions to align with system dynamics, thus refining man-machine interfaces (D.T. McRuer et al. (1967)). These findings emphasize the significance of recognizing human operator dynamics to ensure effective interaction between humans and robots, ultimately enhancing system performance and user satisfaction.

Table 2.1: An overview of the selected articles.

Article	WMM	Smart walker	Cane robot	Falling prevention	Force sensor	Gait estimation/ model	STS	Impedance/ Admittance	Supportive forces
Zhao et al. (2020)		X		X	X	X			
Chang et al. (2021)		X		X	X	X			X
D. Ding et al. (2023)		X		X	X				X
L. Ding et al. (2022)	X			X	X	X		X	
Xing et al. (2021)	X							X	X
Itadera and Cheng (2022)					X	X		X	X
Itadera, Dean-Leon, et al. (2019)		X			X	X		X	X
Mahdi et al. (2022)		X					X		X
Itadera, Kobayashi, et al. (2019)		X			X	X	X	X	X
Yokota et al. (2019)		X		X	X		X	X	X
Naeem et al. (2022)			X	X	X			X	X
B et al. (2023)			X	X	X				
Itadera, Nakanishi, et al. (2020)			X	X	X	X		X	X
Cerqueira et al. (2023)			X	X	X			X	X
Bilgin et al. (2023)		X		X	X	X		X	X
Li, Y. Yamada, Wan, et al. (2019)		X			X	X			X
Li, Y. Yamada, K. Yamada, et al. (2022)		X			X	X			X

3

Trust in Automation

In the field of HRI, robots extend their operations beyond controlled environments, engaging in close collaboration with humans. This transition emphasizes the importance of ensuring safety and trust. Considerations regarding Trust in Automation encompass various concerns, from privacy apprehensions to the ethical implications of assigning tasks to robots. Moreover, it includes the importance of establishing trust between users and robotic systems. The following discussions on Trust in Automation are extracted from the chapter in the project report Langeweg (2023).

One of the main concerns related to HRI with vulnerable users, is ensuring that the robot is designed and programmed to prioritize the safety and well-being of the user. Akalin et al. (2022) indicates that users' perception of safety during HRI is influenced by factors such as the transparency of robot behaviors and predictability. The study emphasizes the importance of developing systems that users, especially vulnerable groups such as older adults, consider safe to interact with. Furthermore, a perception of safety is accomplished by emphasizing the key factors of transparency, predictability, a sense of control and trust.

In the context of service robots in elderly care, these aspects become even more crucial. Transparency allows elderly users to understand the robot's actions and intentions through clear communication and feedback (Akalin et al. (2022)). Predictability is essential for user comfort and safety, as it ensures consistent and reliable robot behavior, allowing users to anticipate its actions and reducing potential distress (Akalin et al. (2022)). Furthermore, granting a sense of control to elderly users empowers them by enabling them to start and stop interactions, preserving their independence and self-assurance. These principles collectively contribute to an environment where elderly users not only receive physical assistance, but also experience a profound sense of trust and control.

An additional ethical consideration is informed consent. Certain vulnerable users may not fully understand what it means to interact with a robot and, therefore, might not be able to give informed consent. It is crucial to obtain consent from the user's legal guardian or caregiver and make sure that the user feels comfortable with the interaction (Akalin et al. (2022)). Additionally, privacy emerges as a significant ethical concern. The design

of the robot must actively respect the user's privacy by not collecting or sharing sensitive information without their explicit consent.

The potential replacement of human interaction with actions carried out by robots, leading to reduced social contact, is a crucial aspect around Trust in Automation. Patients and healthcare professionals fear a loss of human connection and empathy, both fundamental in delivering high quality care (Servaty et al. (2020)).

To tackle these worries, it is vital to involve patients and healthcare workers in the design and implementation of robotic devices. This ensures that robots complement and enhance human interaction instead of substituting it (Akalin et al. (2022)). For instance, robots can handle routine tasks such as monitoring health data or medication delivery, allowing healthcare professionals to spend more time engaging with patients and offering emotional support. Moreover, providing patients with clear information about the role of robots in healthcare and addressing their concerns helps establish trust in the technology (Akalin et al. (2022)).

In conclusion, this chapter highlights the importance of user safety and trust, providing a sense of control in decision-making processes. It emphasizes the necessity of balancing the benefits of robotic assistance and preserving human autonomy to enhance the quality of care.

4

Theory

This chapter focuses on essential concepts to guide the development of a smart walker for elderly care. The central idea is to understand how HRI is enhanced through the implementation of admittance control, emphasizing the importance of compliance. Additionally, the coordinate transformation of a force sensor is explored to unveil a correlation between the sensor data and the interaction coordinates. Lastly, the examination of interaction dynamics between the walker and users informs strategies designed for fall prevention and guiding users along a path. The fundamental theory presented in this chapter originates from the project report (Langeweg (2023)) and has been expanded to offer additional functionalities for the user. Notably, Section 4.1, 4.2, 4.3 and 4.4 are directly extracted from the project.

4.1 Human Robot Interaction

Historically, robots were confined to repetitive tasks with predefined instructions. However, as technology advances, robots engage in increasingly complex and unstructured tasks. Robots are no longer mere automatons executing programmed routines, they are becoming an integral part of our daily lives. As robots evolved, the necessity for HRI emerged (Feil-Seifer et al. (2009)). As defined by ISO, HRI represents the "information and action exchanges between human and robot to perform a task by means of a user interface" (*ISO 8373:2021, Robotics* (2023)). In other words, HRI focuses on understanding and enhancing the relationships between humans and robots. Its core objectives involve developing principles and algorithms for robots to safely and effectively interact with humans in various environments (Feil-Seifer et al. (2009)).

4.2 Compliance

The idea of compliance is crucial in the field of HRI, especially in the context of safety. Compliance, as defined by ISO, is "flexible behavior of a robot or any associated tool in

response to external forces exerted on it" (*ISO 8373:2021, Robotics* (2023)). In essence, the interaction between the user and the robot is dynamic. This idea is applied by smart walkers and assistive devices which offer assistive forces rather than rigorously prescribing motions.

Excessive rigidity in technology might lead to balance issues among elderly individuals. Consequently, integrating compliant behavior into the assistive device has the potential to reduce falls. This is achieved through the application of impedance control or admittance control (Khan et al. (2014)). The resulting interaction improves the senior's autonomy by offering support, while maintaining a sense of control.

According to Hogan (1984) humans adjust the impedance of their arm when they interact with an unpredictable environment. Therefore, the arm's stiffness is adjusted to facilitate a soft or stiff interaction. In this context, impedance defines the relation between force and motion, expressed as

$$F = Z \cdot \dot{x}, \tag{4.1}$$

where F is the contact force, Z the impedance and $\dot{x}$ the velocity.

Correspondingly, the interaction between humans and robots can exploit this principle (Peternel et al. (2017)). The mechanical impedance of a structure is defined as the resistance to motion when an external force is applied (Khan et al. (2014)), while admittance is the inverse of impedance. Thus, high admittance would result in faster motion for a given force in contrast to high impedance.

Furthermore, impedance and admittance encompass several parameters, such as stiffness, damping, inertia, etc., each corresponding to either position or one of its time derivatives. Due to the practical difficulties involved in measuring variables beyond acceleration, impedance and admittance are typically represented by a mass-spring-damper (Peternel et al. (2017)).

The inherent causality in a mechanical system dictates that only forces can excite a system, resulting in a corresponding motion. This fundamental property enables the characterization of a mechanical system through its admittance. As a result, admittance control can assist walking for the elderly by generating compliant behaviors between the user and the robotic system.

4.2.1 Admittance Controller

Admittance control is one of the most common methods for interaction between humans and robots, as it controls the motion of the walker based on the applied force and/or torque (Cerqueira et al. (2023)). Furthermore, the admittance model emulates a dynamic system, making users feel like they are interacting with the system specified by the model. Consequently, by modifying the admittance parameters, the interaction dynamics change.

When designing an admittance controller for a nonholonomic robot, it is essential to consider the imposed velocity constraints (Langeweg (2023)). To address the mechanical limitations that prevent sideways sliding and to constrain the robot's movement, linear position x and heading angle θ serve as the relevant states

$$\mathbf{p} = \begin{bmatrix} x & \theta \end{bmatrix}^T. \tag{4.2}$$

Applying the previously defined coordinates (Equation 4.2), the robot's base is controlled using an admittance controller as follows

$$\mathbf{M}\ddot{\mathbf{p}} + \mathbf{D}\dot{\mathbf{p}} + \mathbf{K}\mathbf{p} = \mathbf{F}, \tag{4.3}$$

where the vector $\mathbf{F} = \begin{bmatrix} f_x & f_y \end{bmatrix}^T$, encapsulates the force exerted by the user in the x- and y-direction. These elements are selected because the force in the x-direction reflects the user's intention to modify linear velocity, while the force in the y-direction corresponds to the user's desire to adjust steering. Consequently, the forward dynamics are decoupled from the turning dynamics, simplifying the detection of the user's intention.

Moreover, $\mathbf{M}, \mathbf{D}, \mathbf{K}$ are the virtual mass, damper and spring coefficient, respectively, defined as

$$\begin{aligned} \mathbf{M} &= \begin{bmatrix} m_x & 0 \\ 0 & m_\theta \end{bmatrix} \\ \mathbf{D} &= \begin{bmatrix} d_x & 0 \\ 0 & d_\theta \end{bmatrix} \\ \mathbf{K} &= \begin{bmatrix} k_x & 0 \\ 0 & k_\theta \end{bmatrix}. \end{aligned} \tag{4.4}$$

Using the established admittance controller, the resulting transfer function from external force $\mathbf{F}$ to position $\mathbf{p}$ is defined as follows:

$$H(s) = \frac{p(s)}{F(s)} = \frac{1}{Ms^2 + Ds + K}. \tag{4.5}$$

4.3 Differential Drive Robot

A differential drive robot utilizes two independently driven wheels located on opposite sides of its body. The robot can adjust its heading by manipulating the relative rotation rates of these wheels, eliminating the need for an additional steering motion (Dudek et al. (2010)). The rotation of the robot occurs around a point along their common axis, referred to as the Instantaneous Center of Rotation (ICR). Dudek et al. (2010) present the geometric relations depicted in Figure 4.1, which implies that the rotation rate ω around the ICR is equal for both wheels.

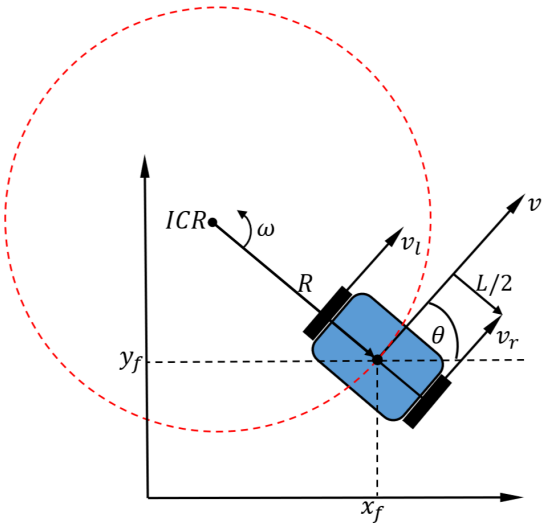


Figure 4.1: Geometric relationship for rotation of a differential drive robot.

Consequently, the following equations are derived

$$\begin{aligned} v_l = (R - \frac{L}{2})\omega \quad &\rightarrow \quad \omega = \frac{v_l}{R - \frac{L}{2}} \\ v_r = (R + \frac{L}{2})\omega \quad &\rightarrow \quad \omega = \frac{v_r}{R + \frac{L}{2}}, \end{aligned} \tag{4.6}$$

where L is the length of the axle.

Utilizing the equality of ω for both wheels results in

$$R = \frac{L}{2}\frac{v_r + v_l}{v_r - v_l}, \tag{4.7}$$

where R represents the signed distance from the ICR to the midpoint of the axle (x_f, y_f).

When $v_r = v_l$, only forward linear motion occurs, causing R to approach infinity. In the case of equal yet opposite velocities in the two wheels, rotation takes place around the midpoint of the wheel axle, resulting in $R = 0$. Additionally, when turning with one wheel at a standstill (e.g., $V_l = 0$), the rotation occurs around that wheel, leading to $R = \frac{L}{2}$.

Moreover, the velocity of the robot traveling tangentially to the path, referred to as the forward linear velocity, is determined by

$$v = \frac{v_r + v_l}{2}. \tag{4.8}$$

4.4 Force Sensor

Force sensing technologies play an important role in enhancing the capabilities of service robots, particularly in applications involving physical interaction with users (Naeem et al.

(2022); Zhao et al. (2020); Chang et al. (2021)). The choice of force sensor depends on the desired accuracy, sensitivity and range of force measurements. As such, sensors that can detect subtle forces are crucial in the context of HRI for elderly care in order to detect user intentions and provide compliant interaction.

4.4.1 Coordinate Transformation

Using a six axes force sensor, it is possible to measure both the force and the torque expressed by

$$\mathbf{T}^f = \begin{bmatrix} \mathbf{F}^f & \mathbf{M}^f \end{bmatrix}^T = \begin{bmatrix} F_{fx} & F_{fy} & F_{fz} & M_{fx} & M_{fy} & M_{fz} \end{bmatrix}^T . \tag{4.9}$$

The force sensor comes with a factory reference frame f, given in Figure 4.2.

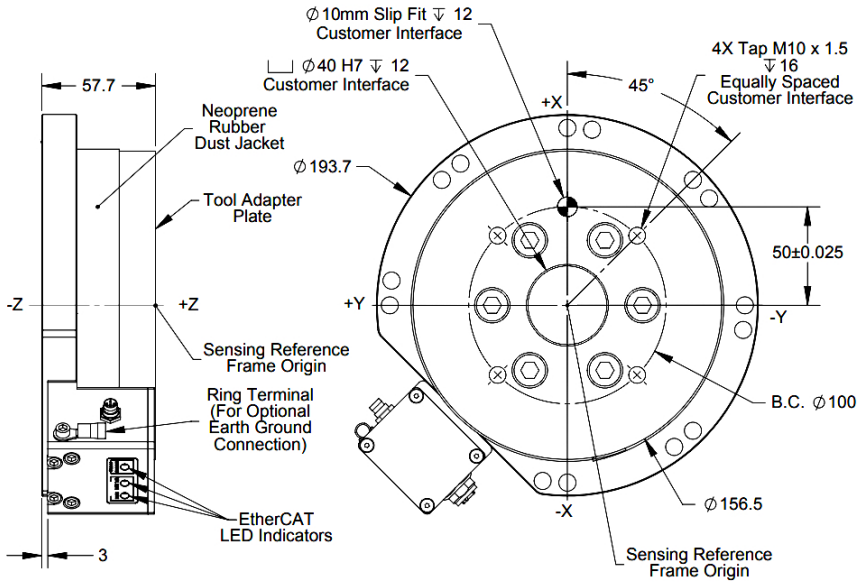


Figure 4.2: Force sensor with factory reference frame.

Due to the placement of the sensor on the robot, there arises a need to rotate the measurement vector 4.9 to the sensor frame s given in Figure 4.3, which is aligned with the handle frames. The corresponding transformation matrix from the factory reference frame f to the sensor coordinates s is expressed as follows:

$$\mathbf{C}_s^f = \begin{bmatrix} \mathbf{R}_s^f & 0 \\ 0 & \mathbf{R}_s^f \end{bmatrix} . \tag{4.10}$$

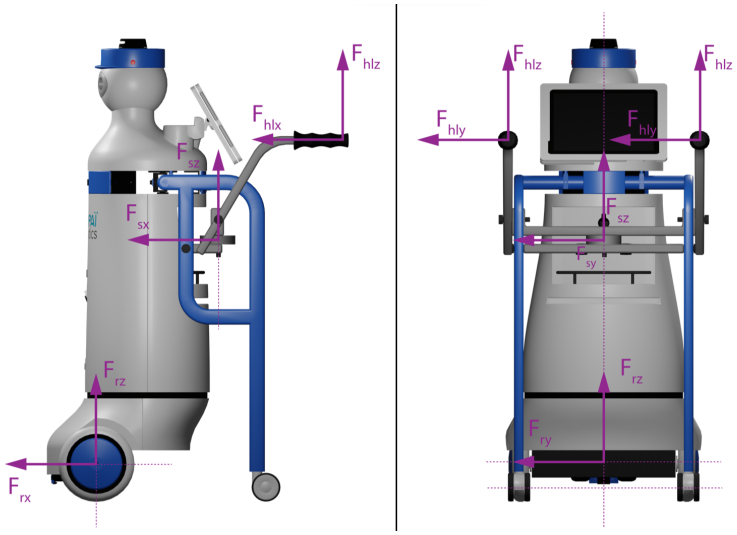


Figure 4.3: Coordinate systems of sensor and handles.

By employing the provided transformation matrix (Equation 4.10), as indicated below (Equation 4.11), the data from the factory reference frame is rotated to attain measurements in the sensor frame that are aligned with the two handles.

$$\mathbf{T}^s = \mathbf{C}_s^f \cdot \mathbf{T}^f \tag{4.11}$$

Obtaining the rotation matrix from the factory frame to the sensor frame involves a sequence of rotations. Initially, there is a rotation of ϕ degrees around the x-axis (roll), followed by θ degrees around the y-axis (pitch) and finally ψ degrees around the z-axis. The rotations are executed counterclockwise relative to their respective axes. The rotation matrices are expressed as follows:

$$\begin{aligned}
\mathbf{R}_x(\phi) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \\
\mathbf{R}_y(\theta) &= \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\phi \end{bmatrix} \\
\mathbf{R}_z(\psi) &= \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}.
\end{aligned} \tag{4.12}$$

A single rotation matrix from the factory frame to the sensor frame can be formed by multiplying the yaw, pitch and roll rotation matrices to obtain

$$\mathbf{R}_s^f(\psi,\theta,\phi) = \mathbf{R}_z(\psi)\cdot\mathbf{R}_y(\theta)\cdot\mathbf{R}_x(\phi) \tag{4.13}$$
$$= \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix}.$$

To achieve alignment between the factory reference frame f of the force sensor (4.2) and the defined sensor frame s, a rotation of 180 degrees around the x-axis, succeeded by a 45-degree rotation around the z-axis, is necessary. Hence, the resulting angles are $\theta = 0, \psi = \frac{\pi}{4}$ and $\phi = \pi$.

By inserting the acquired angles into the analytic expression for the rotation matrix (Equation 4.13), the resulting rotation matrix is derived as

$$\mathbf{R}_s^f = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 \\ \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 0 \\ 0 & 0 & -1 \end{bmatrix}. \tag{4.14}$$

4.4.2 Reconstruction of Handle Forces

Due to the physical interaction between the user and the robot, there are contact forces

$$\begin{aligned} \mathbf{F}^{hl} &= \begin{bmatrix} F_{hlx} & F_{hly} & F_{hlz} \end{bmatrix}^T \\ \mathbf{F}^{hr} &= \begin{bmatrix} F_{hrx} & F_{hry} & F_{hrz} \end{bmatrix}^T, \end{aligned} \tag{4.15}$$

between the environment and the left and right handle attached to the robot. Superscript `hl` means decomposed into the coordinate system x_{hl}, y_{hl}, z_{hl} illustrated in Figure 4.3.

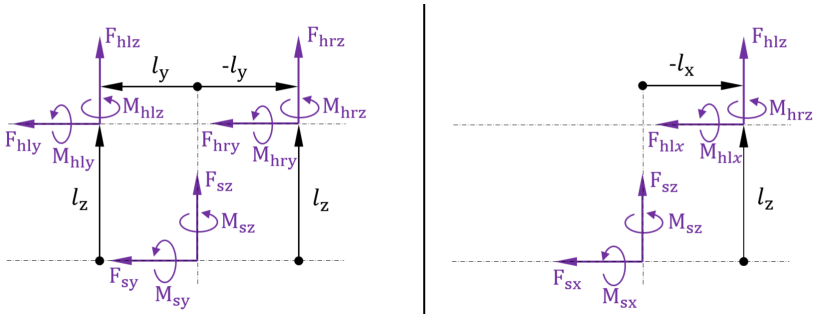


Figure 4.4: Moments and moment arms defined with respect to the sensor origin.

The moments around the sensor frame are generated by the forces applied to the handles. Hence, the moment arms are determined by the distance from the sensor frame to the corresponding handle frame (Figure 4.4):

$$\begin{aligned}\mathbf{D}^{hl} &= \begin{bmatrix}-l_x & l_y & l_z\end{bmatrix}^T \\ \mathbf{D}^{hr} &= \begin{bmatrix}-l_x & -l_y & l_z\end{bmatrix}^T.\end{aligned} \tag{4.16}$$

The moment generated in the sensor frame is expressed by the cross-product of the moment arm and the corresponding forces, as illustrated in the following expression:

$$\begin{aligned}\mathbf{M}^s &= \mathbf{D}^{hl} \times \mathbf{F}^{hl} + \mathbf{D}^{hr} \times \mathbf{F}^{hr} + \mathbf{M}^{hl} + \mathbf{M}^{hr} \\ \mathbf{M}^s &= \begin{vmatrix}\hat{\mathbf{i}} & \hat{\mathbf{j}} & \hat{\mathbf{k}} \\ -l_x & l_y & l_z \\ F_{xhl} & F_{yhl} & F_{zhl}\end{vmatrix} + \begin{vmatrix}\hat{\mathbf{i}} & \hat{\mathbf{j}} & \hat{\mathbf{k}} \\ -l_x & -l_y & l_z \\ F_{xhl} & F_{yhl} & F_{zhl}\end{vmatrix} + \mathbf{M}^{hl} + \mathbf{M}^{hr}.\end{aligned} \tag{4.17}$$

Assuming that the moments applied to the handles by the user, denoted as M^{hl} and M^{hr}, have no impact on the moment in the sensor M^s, the expression is simplified to

$$\begin{bmatrix}M_{sx} \\ M_{sy} \\ M_{sz}\end{bmatrix} = \begin{bmatrix}l_y(F_{hlz} - F_{hrz}) - l_z(F_{hly} + F_{hry}) \\ l_x(F_{hlz} + F_{hrz}) - l_z(F_{hlx} + F_{hrx}) \\ l_y(F_{hrx} - F_{hlx}) - l_x(F_{hly} + F_{hry})\end{bmatrix}. \tag{4.18}$$

Three additional equations are necessary to solve for the handle forces. These equations can be derived from the net force measured in the sensor, which is the sum of both handle forces, expressed as

$$\begin{aligned}F_{sx} &= F_{hlx} + F_{hrx} \\ F_{sy} &= F_{hly} + F_{hry} \\ F_{sz} &= F_{hlz} + F_{hrz}.\end{aligned} \tag{4.19}$$

By substituting the expressions in Equation 4.19 into Equation 4.18 and solving for the handle forces, the result is obtained in Equation 4.20.

$$\begin{aligned}F_{hlx} &= \frac{1}{2}F_{sx} - \frac{1}{2l_y}(M_{sz} + l_x F_{sy}) \\ F_{hrx} &= \frac{1}{2}F_{sx} + \frac{1}{2l_y}(M_{sz} + l_x F_{sy}) \\ F_{hlz} &= \frac{1}{2}F_{sz} + \frac{1}{2l_y}(M_{sx} + l_z F_{sy}) \\ F_{hrz} &= \frac{1}{2}F_{sz} - \frac{1}{2l_y}(M_{sx} + l_z F_{sy})\end{aligned} \tag{4.20}$$

Despite successfully deriving equations for the forces in x- and z-direction, the derivation of forces in the y-direction, denoted as F_{hly} and F_{hry}, faces a challenge. This challenge arises from the negative moments generated by both these forces around the x

and z axes. Consequently, distinguishing between the y-forces applied to the left and right handles becomes infeasible, as both contribute to moments in the same direction.

Nevertheless, the force in the y-direction solely influences turning. Consequently, measuring F_{sy}, which represents the combined forces, is adequate to predict the user's intention to turn.

4.5 Fall Prevention

Identifying the risks of falls is essential for effective fall prevention strategies. Falls among the elderly often originate from a combination of factors, including balance impairment, muscle weakness and cognitive decline (Al-Aama (2011)). These factors create vulnerabilities that increase the risk of falling.

To mitigate fall risks, real-time monitoring and responsive interventions are crucial. Vertical forces are indicative of weight distribution and balance, making them valuable metrics for assessing the risk of falls (Zhao et al. (2020)). Hence, by continuously monitoring vertical forces exerted during walking, the robot can detect deviations from normal patterns that might signal a loss of balance. Detecting potential balance issues and triggering appropriate responses, such as adjusting its support, enables the robotic system to assist the user in regaining stability. This proactive approach contributes to maintaining the stability and safety of elderly users.

The user's relative position to the robot is represented as ${}^{r}x_h$ and ${}^{r}y_h$ in the robot's coordinate system. Utilizing these measurements, the linear velocity of the user can be calculated as follows

$$\dot{x}_h[n] = ({}^{r}x_h[n] - {}^{r}x_h[n-1])\Delta t + v_x, \tag{4.21}$$

where v_x is the linear velocity of the robot and n is the current timestep.

To distinguish between walking and emergency states, the user's relative position to the robot $({}^{r}x_h, {}^{r}y_h)$ is employed. The analysis in Hirata et al. (2006) reveals that the distance distribution of user position frequencies along the x-axis and y-axis during normal walking is a normal distribution. Thus, the user state can be estimated as the Probability Density Function (PDF) of a bivariate normal distribution (Hirata et al. (2006))

$$\begin{aligned} P_{xy} = f({}^{r}x_h {}^{r}y_h) &= \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \exp\left\{ -\frac{1}{2(1-\rho^2)} Q \right\} \\ Q &= \frac{({}^{r}x_h - \mu_x)^2}{\sigma_x^2} - \frac{2\rho({}^{r}x_h - \mu_x)({}^{r}y_h - \mu_y)}{\sigma_x\sigma_y} + \frac{({}^{r}y_h - \mu_y)^2}{\sigma_y^2}, \end{aligned} \tag{4.22}$$

where ρ is the correlation coefficient expressed as follows:

$$\rho = \frac{\sum_{i=1}^{n}({}^{r}x_{hi} - \mu_x) \cdot ({}^{r}y_{hi} - \mu_y)}{\sqrt{\sum_{i=1}^{n}({}^{r}x_{hi} - \mu_x)^2} \cdot \sqrt{\sum_{i=1}^{n}({}^{r}y_{hi} - \mu_y)^2}}. \tag{4.23}$$

The probability of a bivariate normal distribution may be visualized as probability ellipses. These ellipses are characterized by two principal axes in $\mathbb{R}^2$, which indicate the

directions of maximum and minimum variance. These axes are commonly referred to as the semi-major (a) and semi-minor (b) axes (Bishop (2006)). The eigenvalues (u_a, u_b) and eigenvectors (λ_a, λ_b) derived from the covariance matrix

$$\Sigma = \begin{pmatrix} \sigma_x^2 & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma_y^2 \end{pmatrix}, \tag{4.24}$$

represent the variances along the principal axes and the orientation of these axes, respectively (Bishop (2006)). Given positive real valued eigenvalues, the surfaces of the PDF represent an ellipse, with the center at μ, the axes oriented along (u_a, u_b) and with scaling factors in the directions of the axes given by $(\sqrt{\lambda_a}, \sqrt{\lambda_b})$ (Bishop (2006)).

Utilizing the bivariate normal distribution given by Equation 4.22, the objective is to compute the probability expressed as

$$P(-a \leq x - \mu_x \leq a, -b \leq y - \mu_y \leq b) = c, \tag{4.25}$$

in order to determine the region indicative of the user being in a normal state (Hirata et al. (2006)). However, in the absence of an inverse Cumulative Distribution Function (CDF) for the bivariate normal, an approximation for the inverse CDF becomes necessary.

The observations outside the region indicative of the normal state may be categorized as outliers. These observations are characterized by their deviation from the predominant data pattern (Rousseeuw et al. (1990)). A widely adopted method for detecting multivariate outliers is through the employment of the squared Mahalanobis distance (Finch (2012)), represented as

$$D_j^2 = (\mathbf{x_j} - \mu)^\top \mathbf{\Sigma}^{-1} (\mathbf{x_j} - \mu), \tag{4.26}$$

where $\mathbf{x_j}$ denotes a specific observation across the set of n variables, μ is the vector of observation means and Σ is the covariance matrix associated with the n variables.

Consequently, the Mahalanobis distance serves as a measure to characterize points on the surface of an ellipse, providing a metric for distance within the distribution. Specifically, when selecting a random point, it is associated with a probability p of having a squared Mahalanobis distance equal to or smaller than a certain threshold. Moreover, the squared Mahalanobis distance of a Gaussian distribution is known to follow a Chi-Square (χ^2_{df}) distribution where the degrees of freedom (df) are equal to the number of variables (n) in the observation.

Additionally, an ellipse is defined such that the probability of the observed value $\mathbf{x}$ being contained within is $1 - \alpha$ (Equation 4.27). This provides an approximation for estimating Equation 4.25. This approximation is achieved by setting the squared Mahalanobis distance equal to the critical value $Z_c = \chi^2_{n,\alpha}$ obtained with a significance level α (*Applied Multivariate Statistical Analysis* (2024)).

$$P\left(D^2 \leq \chi^2_{n,\alpha}\right) = 1 - \sigma = c \tag{4.27}$$

To determine the $1 - \alpha$ confidence ellipse, the Percent-Point Function (PPF) of the chi-squared distribution is employed. The PPF serves as the inverse of a CDF, providing the critical value ($Z_c = \chi^2_{n,\alpha}$) at which the CDF reaches a specified probability (*1.3.6 Probability Distributions* (2024)). Essentially, this critical value acts as a threshold to ensure that the ellipse covers a certain probability mass. The critical value is applied

to scale the semi-major and semi-minor axes of the ellipse, with the scaling being proportional to the square root of the product of the critical value and the corresponding eigenvalue (Bishop (2006)).

$$\begin{aligned} a &= \sqrt{Z_c \cdot u_a} \\ b &= \sqrt{Z_c \cdot u_b} \end{aligned} \tag{4.28}$$

The derived ellipse is utilized for emergency state estimation. If the user's position falls outside the support ellipse, the walker transitions into an emergency state. To avoid user falls when an emergency state is detected, the damping parameters are increased according to the distance between the walker and the user (Hirata et al. (2006)).

Hirata et al. (2006) modify the damping parameters when in an emergency state, based on the distance d_h and the angle θ_h from the user's position to the mean position of the bivariate normal distribution (μ_x, μ_y).

The user position is measured in the sensor frame as ${}^r x_h, {}^r y_h$ relative to the walker. To determine the distance and angle from the user position to the ellipse center, an inverse rotation is applied to align the coordinate system with the semi-major and minor axes of the ellipse

$$\begin{bmatrix} {}^e x_h \\ {}^e y_h \end{bmatrix} = R(\theta)^{-1} \begin{bmatrix} {}^r x_h \\ {}^r y_h \end{bmatrix}, \tag{4.29}$$

where ${}^e x_h, {}^e y_h$ is the user position in the ellipse frame and R is the rotation matrix

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}. \tag{4.30}$$

This rotation operation becomes relevant when the correlation coefficient ρ is non-zero. In such cases, eigenvalues and eigenvectors corresponding to the semi-major (u_a, λ_a) and semi-minor (u_b, λ_b) axes are computed. The rotation angle θ is determined using the $\arctan 2$ function applied to the eigenvector associated with the major axis, specifically

$$\theta = \arctan 2 \left(\frac{u_{a,y}}{u_{a,x}} \right), \tag{4.31}$$

where $u_{a,y}$ is the y-component and $u_{a,x}$ is the x-component of the eigenvector.

The position in the ellipse coordinate frame can be reformulated by utilizing polar coordinates to determine the distance d_h and angle θ_h from the user to the center of the support ellipse, as expressed by

$$\begin{aligned} d_h &= \sqrt{({}^e x_h)^2 + ({}^e y_h)^2} \\ \theta_h &= \arctan 2 \left(\frac{{}^e y_h}{{}^e x_h} \right). \end{aligned} \tag{4.32}$$

Furthermore, it is necessary to determine the distance between the mean position of the normal distribution, denoted as (μ_x, μ_y), and the edge of the ellipse along the line with

the angle θ_h. This involves identifying the intersection point of the ellipse along the line passing through ${}^{e}x_h, {}^{e}y_h$ and the center of the ellipse. By utilizing the standard ellipse equation

$$\left(\frac{{}^{e}x_h}{a}\right)^2 + \left(\frac{{}^{e}y_h}{b}\right)^2 = 1, \tag{4.33}$$

along with the polar coordinates of the user position ${}^{e}x_h = r_e \cos\theta_h$ and ${}^{e}y_h = r_e \sin\theta_h$, it yields the equation

$$r_e = \sqrt{\frac{1}{\left(\frac{\cos\theta_h}{a}\right)^2 + \left(\frac{\sin\theta_h}{b}\right)^2}}. \tag{4.34}$$

All the steps detailed in this section are outlined in Figure 4.5.

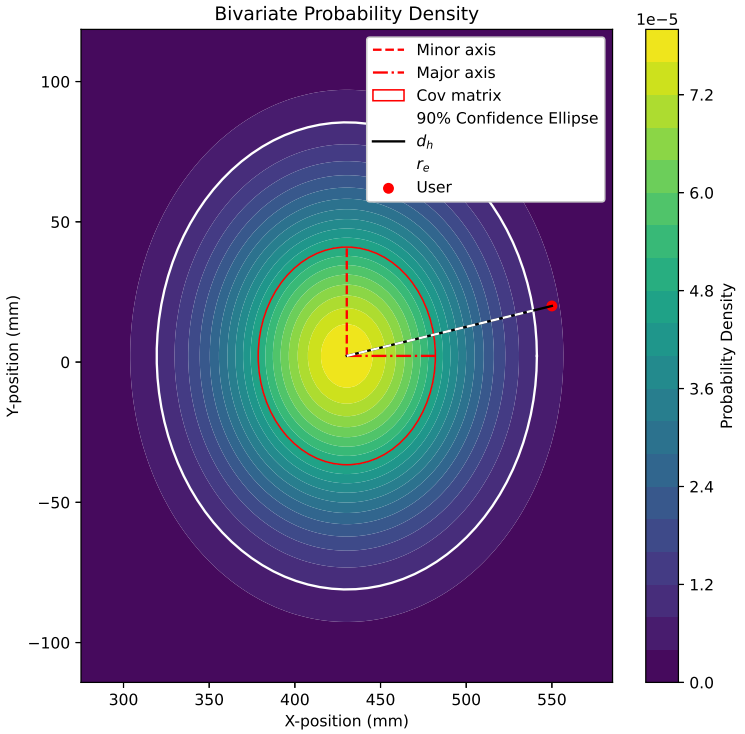


Figure 4.5: Walking ellipse with a 90% confidence interval and distance to user d_h and ellipse r_e.

If the user is situated outside the support ellipse, modifications to the damping coefficients are applied based on the distance from the ellipse $d_h - r_e$ (Bishop (2006))

$$\begin{aligned} d_x &= \exp\{c_x(d_h - r_e)\} + d_{x,\text{init}} \quad (d_h > r_e) \\ d_\theta &= \exp\{c_\theta(d_h - r_e)\} + d_{\theta,\text{init}} \quad (d_h > r_e) \\ d_x &= d_{x,\text{init}} \quad (d_h \leq r_e) \\ d_\theta &= d_{\theta,\text{init}} \quad (d_h \leq r_e), \end{aligned} \tag{4.35}$$

where $d_{x,\text{init}}$ and $d_{\theta,\text{init}}$ represent the initial damping coefficients in the typical walking state, while the values of c_x and c_θ are determined to achieve the desired behavior for fall prevention.

4.6 Guidance

When employing a robot for guidance, a predefined path becomes necessary for the robot to follow. This entails the generation of motion profiles for the chosen control variables. In the context of nonholonomic robots, a common choice of control variables includes the linear velocity $v_d = \dot{x}_d$ and the steering angle δ_d.

However, providing a forward velocity and steering angle introduces challenges in understanding the robot's behavior within the Cartesian space (x, y). Consequently, to obtain a desired behavior in the Cartesian space, a trajectory in the xy-plane is generated and transformed to linear velocity and steering angle (Langeweg (2023)). The robot's spatial configuration and movement are described as follows

$$\begin{bmatrix} x_f & y_f & \theta \end{bmatrix}^T, \tag{4.36}$$

where x_f and y_f represent the position at the center of the front axle along the x- and y-axis, while θ describes the heading angle of the robot in radians.

Conversion between the generated trajectory and the control variables is accomplished by employing a Line of Sight (LoS) controller. The LoS controller was investigated in Langeweg (2023) and the following section is extracted from it.

4.6.1 Line of Sight Controller

An approach for defining a trajectory is through waypoints in the Cartesian space, represented as $(x, y) \in \mathbb{R}^2$. As proposed by Fossen et al. (2003), the path-following problem can be divided into two objectives. The first objective, referred to as "geometric assignment" is to achieve convergence to the desired path. The second objective, known as "speed assignment", is focused on the regulation of the vehicle's speed. Fossen et al. (2003) introduce a LoS controller to reach the first objective. The position, defined as $p = \begin{bmatrix} x & y \end{bmatrix}^T$, converges to the desired path by forcing the heading angle θ to converge to the LoS angle

$$\psi_{los} = \text{atan2}(y_{los} - y, x_{los} - x). \tag{4.37}$$

The geometric relationships are depicted in Figure 4.6.

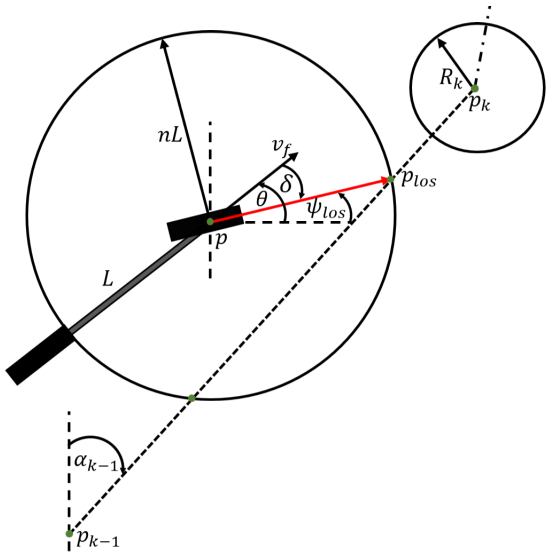


Figure 4.6: Geometric relationship for the LoS controller scheme.

Moreover, the path is defined by a sequence of waypoints, where p_k represents the current waypoint and p_{k-1} denotes the preceding one. By considering the current vehicle position p as the center of a circle with radius nL, where L is the vehicle's length, the circle intersects with the straight-line segment between two consecutive waypoints. The LOS position p_{los} is determined by identifying the closest intersection point to the current waypoint. This position is obtained runtime by solving a system of equations given by

$$\begin{aligned} (y_{los} - y)^2 + (x_{los} - x)^2 &= (nL)^2 \\ \frac{y_{los} - y_{k-1}}{x_{los} - x_{k-1}} &= \frac{y_k - y_{k-1}}{x_k - x_{k-1}} = \tan(\alpha_{k-1}). \end{aligned} \tag{4.38}$$

To ensure that the vehicle successfully traverses the entire path, it is imperative to select the subsequent waypoint. A criterion for selecting the subsequent waypoint, positioned at $p_{k+1} = \begin{bmatrix} x_{k+1} & y_{k+1} \end{bmatrix}^T$, is to ensure that the vehicle is within the radius of acceptance of the current waypoint p_k. Therefore, if, at any given time t, the vehicle's position $p(t)$ satisfies

$$(x_k - x(t))^2 + (y_k - y(t))^2 \leq R_k^2, \tag{4.39}$$

where $R_k < nL$, the next waypoint is selected.

4.6.2 Variable Admittance Control

Haptic feedback provides users with tactile- or force-feedback in response to their interactions with a system or device. This sensation aims to replicate real-world forces relevant to specific tasks (Morris et al. (2007)). By providing virtual overlays with rigid boundaries the user's movements are constrained within defined surfaces (Abbink et al.

(2009)). However, many manual control tasks occur in fluid and unpredictable work environments. In such scenarios, haptic guidance should not impose restrictions but rather emphasize preferred control strategies to enhance performance or user safety (Abbink et al. (2009)).

The controller proposed by Jiménez et al. (2019), denoted as Variable Admittance Control (VAC), introduces haptic sensation by dynamically modifying damping parameters according to user force and deviation from the path. The damping parameter indicates the correct direction by decreasing when the device follows the path closely. These parameters control both linear and angular velocities, thus making it easier to move in the direction of the path.

Furthermore, the orientation error is defined as the difference between the robot orientation θ and the desired steering angle δ_d from the path planner

$$\tilde{\theta} = \delta_d - \theta. \tag{4.40}$$

The damping parameter for the linear velocity d_x has the behavior of an inverted Gaussian. This function offers changes with soft transitions, which are reflected in the user experience (Jiménez et al. (2019)). Thus, the damping parameter d_x is given by

$$d_x = d_{x,max} - d_{d,max} \cdot \exp\left(-\left(\frac{\tilde{\theta}}{\delta_{\mathrm{dx}}}\right)^2\right), \tag{4.41}$$

where, $d_{x,max}$ is the maximum limit of d_x, $d_{d,max}$ is the maximum decrease of damping and δ_{dx} is the parameter that determines the width of d_x function.

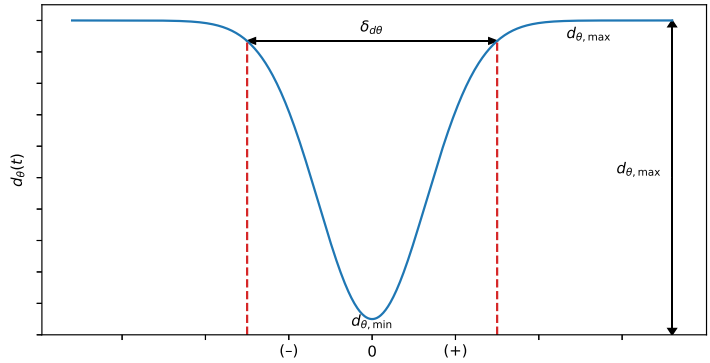


Figure 4.7: Description of constants in the damping parameter for the linear velocity.

Consequently, when $\tilde{\theta}$ approaches zero, the damping is at its minimum, facilitating smooth movement. However, as the orientation error increases, locomotion becomes increasingly challenging for the user.

Moreover, the damping parameter for angular velocity d_θ is given by

$$d_\theta = d_{i\theta} + G_{d\theta} \cdot \tanh\left(\frac{1}{P_{d\theta}} F_y \tilde{\theta}\right), \tag{4.42}$$

where $d_{i\theta}$ is the initial damping parameter, $G_{d\theta}$ the difference between the max and min desired damping parameter, and $P_{d\theta}$ is the slope of the curve of d_θ.

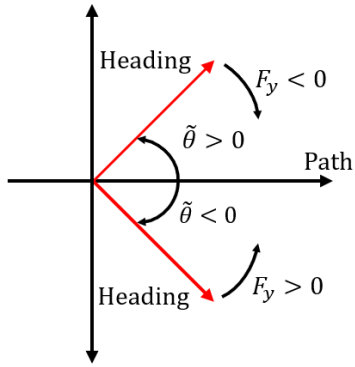


Figure 4.8: Description of constants in the damping parameter for the angular velocity.

When the orientation error $\tilde{\theta}$ is positive, the damping parameters decrease in response to a negative user force F_y, indicating the user's intention to correct the error. Conversely, a positive user force increases the damping parameters, constraining walker mobility. On the contrary, when the error $\tilde{\theta}$ is negative, a positive user force reduces the damping parameter.

The utilization of adaptive damping parameters facilitates haptic feedback. This approach dynamically adjusts parameters in response to user force and deviations from intended paths, thereby offering intuitive guidance without imposing rigid constraints.

4.7 Performance Metrics

This section investigates a variety of metrics to assess the performance of the system. These metrics are essential for evaluating system effectiveness and the maneuverability of the robot.

4.7.1 Physiological Cost Index

Adjusting the virtual coefficients of the admittance controller allows for customization of both the velocity and the walking load to suit the user. Itadera, Nakanishi, et al. (2020) examine the correlation between physiological cost and admittance parameters. One approach to assess cost involves the Physiological Cost Index (PCI) as a measure of energy consumption during walking. The PCI (Itadera, Nakanishi, et al. (2020)) is defined as

$$\mathrm{PCI} = \frac{\mathrm{MeanHR\ at\ walk} - \mathrm{MeanHR\ at\ rest}}{\mathrm{Walking\ speed\ [m/min]}}, \tag{4.43}$$

where HR is the heart rate. The PCI is utilized as a means of estimating energy consumption during walking, where a larger PCI indicates greater energy consumption (Itadera, Nakanishi, et al. (2020)).

4.7.2 Work and Impulse

Additionally, Itadera, Nakanishi, et al. (2020) examines the total work performed by the user on the robot and the impulse transferred from the user to the robot. These measures are computed based on the force applied to the handle and the distance walked, as follows

$$\begin{aligned} W &= W_x + W_y = \int_0^d f_{sx}(t)\, dx + \int_0^d f_{sy}(t)\, dy \\ I &= I_x + I_y = \int_0^{T_c} \left(f_{sx}(t) + f_{sy}(t)\right)\, dt, \end{aligned} \tag{4.44}$$

where d is the distance walked and T_c the duration.

However, these metrics are tailored for a cane-type robot equipped with omnidirectional wheels. In the case of a nonholonomic robot, which lacks the ability to move in any direction, the force in the y-direction alone is not sufficient to induce movement along the y-axis. Therefore, it is more appropriate to assess the resultant force generated by both the x- and y-forces, along with the resultant displacement formed by combining Δx and Δy. By discretizing the integral, the following equation is derived

$$\begin{aligned} W &\approx \sum_i \left(\sqrt{f_{sx}[i]^2 + f_{sy}[i])^2} \cdot \sqrt{\Delta x[i]^2 + \Delta y[i]^2} \right) \\ I &\approx T_c \cdot \sum_i \left(\sqrt{f_{sx}[i]^2 + f_{sy}[i])^2} \right). \end{aligned} \tag{4.45}$$

The calculation of the impulse is straightforward. However, the dissimilar update rates of the force sensor ($f = 500Hz$) and the position measurement ($f = 20Hz$) present a challenge to the discretization process of the work metric, as each timestep i does not align for both measures. To address this issue, one solution involves weighting all force measurements by a fraction of the total resultant displacement. This is achieved by applying the trapezoidal rule, defined as

$$\int_a^b f(x)dx \approx \frac{h}{2} \left[f[a] + 2 \sum_{i=1}^{N_s - 1} f[i] + f[b] \right], \tag{4.46}$$

to the function

$$f[i] = \sqrt{f_{sx}[i]^2 + f_{sy}[i]^2}, \tag{4.47}$$

where $f_{sx}[i]$ and $f_{sy}[i]$ denote the force measurement at timestep i over a horizon N_s. Moreover, the weight of each sub-interval is given by

$$h = \frac{1}{N_s} \sum_{j=0}^{N_d - 1} \sqrt{\Delta x[j]^2 + \Delta y[j]^2}, \tag{4.48}$$

where $\Delta x[j]$ and $\Delta y[j]$ denote the displacement measurement at timestep j over a horizon $N_d \neq N_s$.

4.7.3 Task Completion Time

Moreover, another metric used to quantify effectiveness and performance is task completion time, which refers to the duration taken by participants to complete predefined walking tasks. Shorter completion times typically correspond to better performance, demonstrating the efficacy of individuals or systems in accomplishing objectives (Munoz Ceballos et al. (2010)). Completion time is defined as

$$T_c = t_e - t_s, \tag{4.49}$$

where t_e represents the time upon task completion, and t_s denotes the time when the task is initiated.

4.7.4 Cross-Track Error

The robot's maneuverability is associated with its deviation from the intended path. Therefore, a metric is employed to quantify the maneuvering performance based on the deviation from the planned route (Steinfeld et al. (2006)). A method to quantify path deviation is by employing the Cross-Track Error (XTE). This error indicates the difference in the predefined path's position and the robot's actual location, projected onto a vector perpendicular to the predefined path (Mondoloni et al. (2005)).

When given a path with straight line segments connecting waypoints, the equation of the current line segment is given by:

$$y = c(x - x_c) + y_c, \tag{4.50}$$

where (x_c, y_c) represents the coordinates of the current waypoint and c denotes the slope of the line, calculated as

$$c = \frac{y_c - y_p}{x_c - x_p}, \tag{4.51}$$

with (x_p, y_p) being the coordinates of the previous waypoint. The slope of the perpendicular line is determined by

$$c_\perp = -\frac{1}{c}. \tag{4.52}$$

Now, considering (x_f, y_f) as the coordinates of the robot's front axle, the perpendicular line passing through the robot's location can be expressed as

$$y = c_\perp(x - x_f) + y_f. \tag{4.53}$$

In order to calculate the cross-track error, it is necessary to identify the intersection point between the path's line segment (Equation 4.50) and the perpendicular line (Equation 4.53). Setting these equations equal to each other results in

$$c(x - x_c) + y_c = c_\perp(x - x_f) + y_f. \tag{4.54}$$

Furthermore, solving for x and y yields

$$\begin{aligned} x &= \frac{c \cdot x_c - c_\perp \cdot x_f + y_f - y_c}{c - c_\perp} \\ y &= c(x - x_c) + y_c. \end{aligned} \tag{4.55}$$

The cross-track error is then determined by the hypotenuse of the differences in the x- and y- coordinates

$$\begin{aligned} dx &= x_f - x \\ dy &= y_f - y, \end{aligned} \tag{4.56}$$

resulting in the cross-track error

$$e = \sqrt{dx^2 + dy^2}. \tag{4.57}$$

However, in cases where the line segment is either vertical or horizontal, the slope c tends toward infinity or zero, respectively. As a result, the described method becomes infeasible. In such cases, the cross-track error is simply the deviation in x-coordinates or y-coordinates, respectively

$$e = \begin{cases} |x_c - x_f| & : c = \infty \\ |y_c - y_f| & : c = 0 \end{cases} \tag{4.58}$$

4.7.5 Velocity Variability

Hirata et al. (2006) utilize the user's walking velocity (Equation 4.21) to estimte a stopped state. This walking velocity is derived from the combination of the user's leg velocity relative to the robotic walker and the walker's velocity. Therefore, the system is considered stopped when the combined velocities converge to zero.

Accordingly, alongside the metrics already presented, an additional metric is proposed to assess the disparity between the velocities of the user and the walker. Substantial deviations in these velocities may imply conflicting efforts between the human and the robot, thus enabling an evaluation of the system's performance based on their correlation. This correlation is defined as

$$\text{Correlation}(V_h, V_x) = \frac{\sum_{i=1}^{n}(V_{h,i} - \bar{V_h})(V_{x,i} - \bar{V_x})}{\sqrt{\sum_{i=1}^{n}(V_{h,i} - \bar{V_h})^2} \cdot \sqrt{\sum_{i=1}^{n}(V_{x,i} - \bar{V_x})^2}}, \tag{4.59}$$

where V_h and V_x represent the vectors containing all velocity measurements over a horizon n for the human and the robot, respectively, with $\bar{V_h}$ and $\bar{V_x}$ denoting their respective means.

5

Kompaï's Hardware and Software

The present chapter expands on the specialization project (Langeweg (2023)), in which Section 5.1 and 5.2 are extracted from it. This chapter outlines essential aspects of the experimental setup involving the Kompaï robot, which is employed to investigate applications of force control in assistive devices. A comprehensive understanding of both hardware and software functionalities is crucial for developing the proposed solution, detailed in the following chapter.

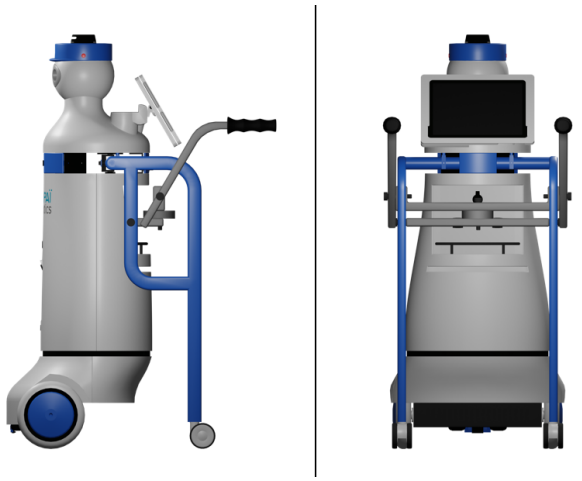


Figure 5.1: The Kompaï robot and its main robot unit and walker section.

5.1 Robot-to-Walker Joint

The Kompaï robot consists of the main robot unit and an attached walker section with handles. The main robot is powered by a differential drive mechanism. Conversely, the

walker part is passive, following the movement dictated by the main robot's actuation. As depicted in Figure 5.1, the connection between the two components is a hinge joint allowing rotation about the vertical axis, enabling the walker to rotate around the robot's body. However, for effective force control, the joint is locked, aligning both components to maintain structural rigidity (Langeweg (2023)).

5.2 Castor Wheels

The Kompaï robot is equipped with swivel castor wheels (Figure 5.2) located at the base of its walker component. These wheels are attached to a fork, with an additional swivel joint above the fork that enables it to freely rotate about 360 degrees (Batlle et al. (2009)). This design facilitates smoother turns, as the wheels of the walker follow the robot's direction. However, during the transition from forward to backward motion or vice versa, the castor wheels undergo a 180-degree rotation, introducing additional friction to the movement. In conclusion, the castor wheels contribute to the robot's enhanced maneuverability, allowing for efficient navigation and directional changes (Langeweg (2023)).

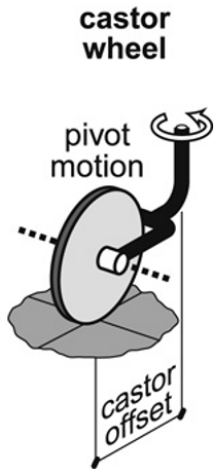


Figure 5.2: Illustration of a castor wheel retrieved from Batlle et al. (2009).

5.3 Direct Differential Actuation

When designing a motion generator, it is crucial to note that the robot employs a differential drive mechanism, where the movement relies on the independent control of its two wheels. The Kompaï robot offers an interface through the open-source software Robot Operating System (ROS) (*ROS: Home* (2023)), allowing for direct control of each wheel.

Consequently, the ability to control the turning radius while preserving the desired forward velocity becomes achievable by independently applying the combined contribution from forward velocity and angular velocity to each wheel. For instance, executing a turn

around one wheel with a constant forward velocity involves keeping one wheel stationary while the other continues at the desired speed. This enables the generation of a smoother motion, thereby enhancing the overall user experience (Langeweg (2023)).

5.4 Kompaï's Path Planner

The Kompaï robot employs two laser range scanners to generate a map of the environment and execute obstacle avoidance. Furthermore, a third laser range finder, positioned at knee height and directed toward the user, detects the distance between the user and the robot.

Moreover, the Kompaï software includes functions for navigating to destinations with or without user support. In these functions, Kompaï's path planner calculates a route for the robot based on the current map. By iterating through waypoints and commanding the robot to reach them, a route can be followed.

6

Methodology

This chapter provides insights into the methodology employed to develop the control system for the Kompaï robot, with a focus on walking assistance, guidance and fall prevention. The primary objective is to facilitate user interaction through a dynamic and responsive robotic system.

6.1 Walking Assistance

Walking assistance, as defined initially, refers to when the service robot provides assistive forces as users command motions. This section will provide a brief overview of the approach outlined in the specialization project (Langeweg (2023)), serving as a reference for comparison with the walking assistance proposed in this thesis.

6.1.1 Turning Strategies

The project report Langeweg (2023), upon which this thesis is built, implemented walking assistance using a Finite State Machine (FSM) and a velocity controller. The velocity controller adjusted the velocity based on the current state of the system and the force applied by the user, thereby providing increments in the velocity during walking. Subsequently, a concise overview of the project's solution will be presented for comparison.

State Machine

The FSM, as shown in Figure 6.1, functions as a framework that updates the robot's state according to force measurements along the x- and z-direction. Establishing the force thresholds in the x-direction as a_x, y-direction as b_y and in the z-direction as c_z, the resulting state machine and state transitions are defined.

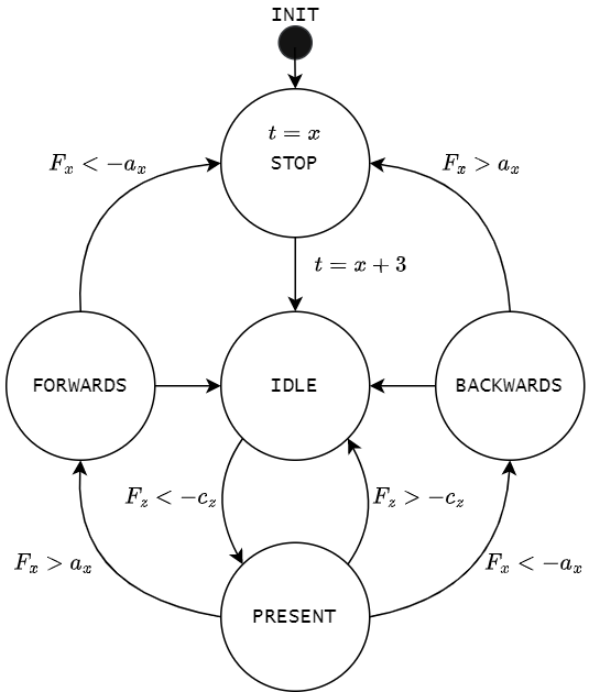


Figure 6.1: The FSM developed in Langeweg (2023).

Furthermore, an algorithm was designed to update the state and reference velocities according to the user applied force. Adjustments to the velocity were controlled with acceleration and deceleration phases with known velocity profiles. Additionally, the ability to initiate turns was facilitated by monitoring the forces exerted on the handles in the y-direction. However, the turning was controlled with a constant angular velocity due to the constraints imposed by the interface of the Kompaï robot.

Due to the recent access to direct control of each wheel, an attempt to control the steering radius is performed in the initial stages of the thesis.

Turning with a Differential Drive Robot

As illustrated in Figure 6.2, the process of turning the robot using a force F_{hly}, assuming $F_{hry} = 0$, entails distinct sign configurations for ω and R depending on the direction of the robot's movement. Table 6.1 provides an overview of these configurations.

	Forward	Backward
$F_{sy} > 0$	$R, \omega < 0$	$R > 0, \omega < 0$
$F_{sy} < 0$	$R, \omega > 0$	$R < 0, \omega > 0$

Table 6.1: Sign configuration for turning when moving forward and backward, respectively.

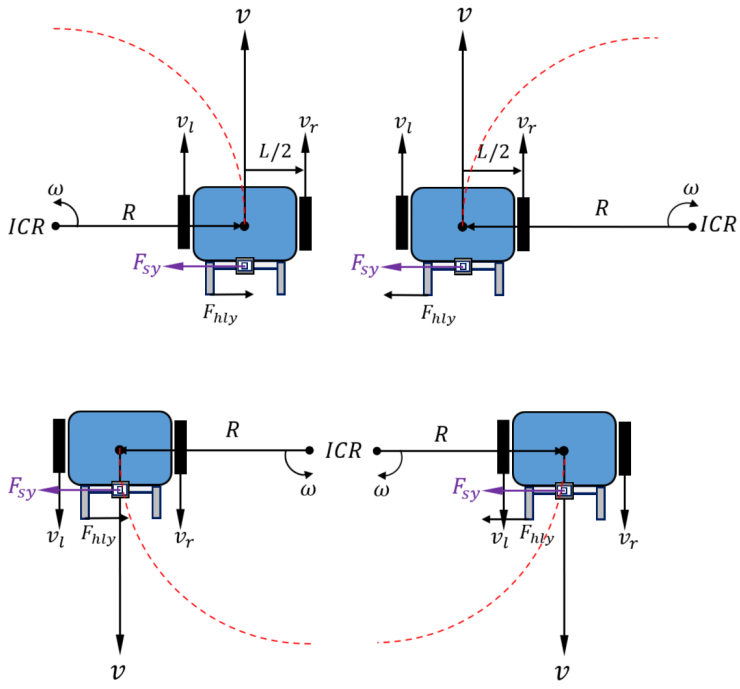


Figure 6.2: Turning with a differential drive robot as a result of the user-applied force F_{hly}.

Since it is infeasible to directly acquire the handle forces in the y-direction, turning is facilitated by monitoring the combined force F_{sy} (Equation 4.19) obtained in the sensor frame. This setup becomes relevant when analyzing user intentions through applied forces.

As such, when the user applies a force $F_{sy} > b_y$, it indicates an intention to turn right, whereas a force $F_{sy} < -b_y$ indicates an intention to turn left. Moreover, turning the robot involves rotating around the vertical z-axis. In accordance with the right-hand rule, a negative angular velocity $\omega < 0$ enables a right turn, whereas a positive angular velocity $\omega > 0$ facilitates a left turn (Langeweg (2023)).

Consequently, the following equation is derived from Table 6.1 to obtain the correct sign configuration

$$\begin{aligned} R &= -\text{sign}(F_{sy}) \cdot \text{dir} \cdot R \\ \omega &= -\text{sign}(F_{sy}) \cdot \omega, \end{aligned} \tag{6.1}$$

where dir is the direction the robot is moving in

$$\text{dir} = \begin{cases} 1 & : \text{State} = \text{FORWARD} \\ -1 & : \text{State} = \text{BACKWARD}. \end{cases} \tag{6.2}$$

Moreover, the right-hand side variables R and ω represent the unsigned values of their respective parameters. The turning radius R is scaled by the exerted force and ω calculated

by one of the three turning strategies: constant ω, constant v, or keeping a constant velocity of the outer wheel.

R scaled by exerted force

The selection of the turning radius R is achieved by scaling it with the magnitude of the exerted force. This is realized by defining parameters $F_{\min}$, $F_{\max}$, $R_{\min}$, and $R_{\max}$, and subsequently deriving R as a linear function

$$R(|F_y|) = (\frac{R_{\min} - R_{\max}}{F_{y,\max} - F_{y,\min}})(|F_y| - F_{y,\min}) + R_{\max}, \tag{6.3}$$

with the saturation

$$R(|F_y|) = \max(\min(R, R_{\max}), R_{\min}), \tag{6.4}$$

in order to achieve $R_{\min} \leq R \leq R_{\max}$.

Additionally, the correct sign for turning is achieved by applying Equation 6.1 to the obtained radius R.

Constant v

The initial strategy for executing a turn involves adjusting the turning radius R based on the user-applied force, while simultaneously preserving a constant forward velocity v. Utilizing the equation for angular velocity

$$\omega = \frac{v}{R}, \tag{6.5}$$

and employing Equation 4.6, the velocities of the left and right wheels are computed as follows

$$\begin{aligned} v_l &= \omega \cdot (R - L/2) \\ v_r &= \omega \cdot (R + L/2). \end{aligned} \tag{6.6}$$

Constant ω

An alternative turning strategy involves utilizing the scaled value of R while maintaining a constant angular velocity ω. This is achieved by selecting a constant $\omega = \omega_d$ and applying Equation 6.6 to get the left and right wheel velocities.

It is important to note that the sign configuration described in Equation 6.1 is applied to the constant ω_d, resulting in:

$$\omega = -\text{sign}(F_{sy}) \cdot \omega_d. \tag{6.7}$$

Constant outer wheel velocity

The final method maintains the forward velocity of the outer wheel while adjusting the

velocity of the inner wheel to achieve the desired turning radius R. Utilizing Equation 6.6 and substituting ω with the equation

$$\omega = \frac{v_l}{R - \frac{L}{2}} = \frac{v_r}{R + \frac{L}{2}}, \tag{6.8}$$

derived in Equation 4.6, leads to the derivation of the following rule

$$\begin{aligned} R \geq 0: \quad & \begin{cases} v_r = v, \\ v_l = v_r \cdot \dfrac{(R - \frac{L}{2})}{(R + \frac{L}{2})} \end{cases} \\ R < 0: \quad & \begin{cases} v_r = v_l \cdot \dfrac{(R + \frac{L}{2})}{(R - \frac{L}{2})}, \\ v_l = v \end{cases} \end{aligned} \tag{6.9}$$

where $R \geq 0$ requires the preservation of the right wheel velocity, while $R < 0$ necessitates the preservation of the left wheel velocity.

Selecting parameters

The process of selecting the turning radius R for the differential drive robot involves careful consideration of the defined parameters. To determine $R_{\min}$, it is crucial to ensure that neither wheel stops nor moves backward during the robot's turn, as this would necessitate the user to step sideways. This precaution is essential as sideways rollover is the most frequent fall type among older adults (Bilgin et al. (2023)). Hence, by minimizing lateral movement, the risk of falling decreases. Therefore, it is recommended to set $R_{\min}$ greater than $\frac{L}{2}$ + margin, where a larger margin places the minimum turning radius further away from the inner wheel. On the other hand, determining $R_{\max}$ involves considering the available space, emphasizing the importance of avoiding an excessively large turning radius.

Additionally, establishing a minimum lateral force threshold $F_{y,\min} = b_y$, is essential to maintain stability during turning maneuvers. If b_y is set too high, oscillatory behavior may occur, as the force will drop below the threshold once the turn is initiated. Conversely, if b_y is set too low, the turn will be initiated with minimal lateral forces, making it challenging to maintain straight movement with the robot. Furthermore, the selection of $F_{y,\max}$, the maximum lateral force, could potentially be guided by practical considerations such as the maximum lateral push. The present approach ensures a set of turning parameters tailored to the robot's dynamics and the user interaction.

6.1.2 Admittance Controller

While the methodology employed in the specialization project (Langeweg (2023)) was rule-based, this thesis aims to develop a more dynamic interaction that complies with user forces. This section describes the implemented admittance controller designed to facilitate dynamic walking assistance.

User Intention

Understanding user intention and responding accordingly is fundamental in the development of effective HRI systems (Wakita et al. (2013)). By accurately interpreting user intentions, robots can provide more intuitive and efficient assistance, enhancing user experience and overall system performance. Establishing clear rules and conditions for detecting user intention is crucial to ensure that the robot responds appropriately to user commands. These rules serve as a foundation for creating dynamic and responsive interaction strategies, enabling the robot to adapt its behavior based on user input.

An important aspect is discerning whether the user is present at the handles. Presence at the handles indicates the user's readiness to interact with the robot and initiate movement. Consequently, releasing the wheel brakes is necessary to facilitate a smoother initiation of movement when the user applies force. To determine user presence, an approach is to evaluate the magnitude of the force vector $\vec{d}$, which is derived as the resultant force from the x- and z-forces

$$||d|| = \sqrt{\vec{F_x}^2 + \vec{F_z}^2} > c, \tag{6.10}$$

where c is a constant threshold that indicates user presence when $||d||$ exceeds the threshold.

When approaching the robot, users typically apply downward pressure on the handles for support. However, as they prepare to initiate motion, they start pushing forwards, potentially causing the handle forces in the z-direction to diminish. To prevent oscillation caused by sudden changes in force, it's necessary to consider both x- and z-directions in the initiation condition, as opposed to solely relying on the z-direction.

Furthermore, to facilitate forward movement, the robot should initiate linear motion if the sum of the horizontal forces on both handles exceeds a certain threshold a_x

$$|F_{hlx} + F_{hrx}| > a_x. \tag{6.11}$$

Under these conditions, an admittance controller regulates the linear velocity, enabling a smooth and dynamic motion.

Similarly, when the user intends to initiate a turn, the robot detects this based on the magnitude of the lateral force exerted on the handles. As introduced previously (section 6.1.1), a lateral force F_{sy} surpassing the predetermined threshold b_y indicates an intention to turn, with the direction determined by the sign of F_{sy}

$$|F_{hly} + F_{hry}| = |F_{sy}| > b_y. \tag{6.12}$$

Under these conditions, the angular velocity is adjusted in accordance with the admittance controller to ensure a smooth and precise execution of the turn.

All the presented conditions are outlined in Figure 6.3, along with various conditions for fall prevention, which will be elaborated on in the following section.

Action	Interpretation	Reaction
$\|\|d\|\| = \sqrt{{f_{sx}}^2 + {f_{sz}}^2} > c$	User present	Release breaks
$\|f_{hlx} + f_{hrx}\| > a_x$	Move forward	Forwar velocity v_x from admittance controller
$f_{sy} = f_{hly} + f_{hry} > b_y$	Turn left	Angular velocity ω from admittance controller
$f_{sy} = f_{hly} + f_{hry} < -b_y$	Turn right	Angular velocity ω from admittance controller
$\|f_{hlz} - f_{hrz}\| > c_{\Delta z}$	Imbalance	Transition to Stop state
$\|f_{hlz} + f_{hrz}\| > c_z$	Imbalance	Transition to Stop state
$d_h > r_e$	Reduced walking support	Adjust admittance parameters d_x and d_θ

Figure 6.3: A summary of user intentions and robot response.

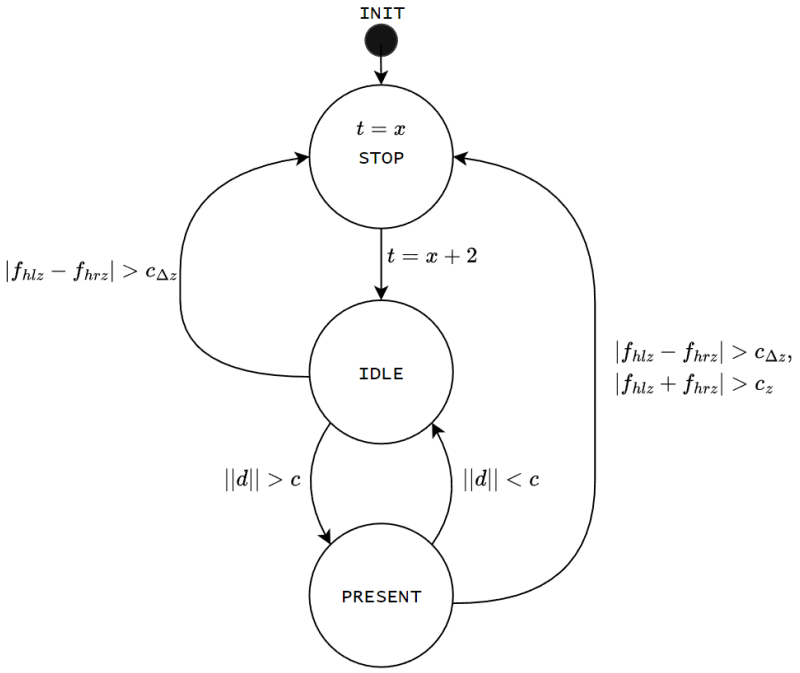


Figure 6.4: The FSM governing state transitions for the walking assistance developed in this thesis.

To adapt to a variety of user inputs, the robot operates with three distinct states governed by a FSM (Figure 6.4). As the walking assistance initiates, the robot enters the STOP state. Following a predefined interval, it transitions to the IDLE state, where user inputs are processed. Furthermore, if the condition specified by Equation 6.10 is met, the system transitions to the PRESENT state and the motor breaks are released. Subsequently, the motion of the robot is governed by the admittance controller where user-applied forces serve as input. Additionally, the system returns to the IDLE state if the condition specified by Equation 6.10 is no longer satisfied, or it reverts to the STOP state in case of an emergency.

Discrete Admittance Model

In the theoretical framework, Equation 4.3 introduced a general admittance model comprising a mass-spring-damper. However, when considering velocity control, incorporating a spring becomes impractical. Springs introduce a restoring force, establishing an additional equilibrium point at the zero position. Consequently, including a spring would result in the walker being drawn back to the initial position, thus impeding free movement.

For that reason, a mass-damper model is considered for the admittance controller. Removing the spring from Equation 4.3, yields

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{D}\mathbf{v} = \mathbf{F}, \tag{6.13}$$

where $\dot{\mathbf{p}}$ is substituted with $\mathbf{v}$, defined as

$$\mathbf{v} = \dot{\mathbf{p}} = \begin{bmatrix} v_x & \omega \end{bmatrix}^T, \tag{6.14}$$

for simpler notation.

Furthermore, by utilizing an explicit integration method such as Forward Euler discretization

$$y[k+1] = y[k] + \Delta t \cdot f(t[k], y[k]), \tag{6.15}$$

a discrete-time model (Equation 6.16) is derived.

$$v[k] = (I - M^{-1}D\Delta t)v[k-1] + M^{-1}F[k-1]\Delta t \tag{6.16}$$

When applying the discrete-time model, the resulting output comprises linear forward velocity and angular velocity. However, converting these velocities into left and right wheel velocities is required to command the robot.

Substituting the equation for the turning radius R (Equation 4.7) and the equation for the forward linear velocity v_x (Equation 4.8) into the equation for angular velocity (Equation 6.5) results in

$$\omega = \frac{v_x}{R} = \frac{v_r + v_l}{2} \cdot \frac{2}{L}\frac{v_r - v_l}{v_r + v_l} = \frac{v_r - v_l}{L}. \tag{6.17}$$

Additionally, by rearranging v_l from Equation 4.8 as

$$v_l = 2v_x - v_r, \tag{6.18}$$

and substituting it into Equation 6.17, yields

$$\omega = \frac{v_r - 2v_x + v_r}{L}. \tag{6.19}$$

By solving Equation 6.19 for v_r and subsequently substituting it into Equation 6.18, the equations for the left and right wheel velocities are derived as

$$\begin{aligned} v_l &= v_x - \frac{L \cdot \omega}{2} \\ v_r &= v_x + \frac{L \cdot \omega}{2}. \end{aligned} \tag{6.20}$$

Commanded Velocity

Using the established admittance controller (Equation 6.13), the resulting transfer function from external force **F** to velocity **v** is defined as follows:

$$H(s) = \frac{v(s)}{F(s)} = \frac{1}{Ms + D}. \tag{6.21}$$

The inverse Laplace transform of Equation 6.21 (Nakagawa et al. (2015)) results in

$$\mathbf{v}(t) = \dot{\mathbf{p}}(t) = \int_0^t G(\tau) \cdot F(t - \tau)\, d\tau, \tag{6.22}$$

where

$$G(\tau) = \begin{bmatrix} \frac{e^{-\frac{\tau}{T_{c,x}}}}{m_x} & 0 \\ 0 & \frac{e^{-\frac{\tau}{T_{c,\theta}}}}{m_\theta} \end{bmatrix}, \tag{6.23}$$

and $T_{c,x}$, $T_{c,\theta}$ are the time constants defined by

$$\begin{aligned} T_{c,x} &= \frac{m_x}{d_x}, \\ T_{c,\theta} &= \frac{m_\theta}{d_\theta}. \end{aligned} \tag{6.24}$$

Since $H(s)$ (Equation 6.21) is a first-order lag element (Nakagawa et al. (2015)), an impulse response converges to zero with time. However, to obtain position, an integrator is added to the transfer function

$$H(s) = \frac{p(s)}{F(s)} = \frac{1}{s(Ms + D)}, \tag{6.25}$$

resulting in an additional pole in the origin. Consequently, the position will not converge to zero with time.

Furthermore, when a constant force f^* is applied, the steady-state value of the forward linear velocity v^* is achieved by rewriting Equation 6.16 as

$$v^* = (I - M^{-1}D\Delta t)v^* + M^{-1}f^*\Delta t, \tag{6.26}$$

and solving it in terms of v^*:

$$v^* = D^{-1}f^*. \tag{6.27}$$

Selecting parameters

Since the admittance parameters determine the interaction dynamics between the user and the robot, they play a crucial role in the overall performance of the system. The initial approach to select parameters involved identifying a relationship between the mass and damper parameters arising from the discretization process.

By examining Equation 6.16, it becomes evident that the expression $(I - M^{-1}D\Delta t)$ must remain non-negative to prevent oscillations. This leads to the following relationship

$$M^{-1}D\Delta t < I, \tag{6.28}$$

ensuring that the velocity does not oscillate between positive and negative values. Moreover, if $(I - M^{-1}D\Delta t) < -I$, equivalent to

$$M^{-1}D\Delta t > 2I, \tag{6.29}$$

the velocity will be amplified, resulting in an unstable system.

Consequently, the selection of admittance parameters relies on the update rate of the admittance controller Δt. The update rate is chosen based on the bandwidth required for human interaction, as delays can significantly impact human performance (Steinfeld et al. (2006)).

Furthermore, the mass parameter is selected based on the desired contribution from one unit of force. The force vector F is scaled by $\Delta t M^{-1}$, resulting in velocity

$$M^{-1}F\Delta t \quad : \quad \frac{1}{\mathrm{m}} \cdot \frac{\mathrm{m}}{\mathrm{s}^2} \cdot s = \frac{\mathrm{m}}{\mathrm{s}}. \tag{6.30}$$

Hence, with one Newton of force, the change in velocity is expressed as

$$\Delta t M^{-1}. \tag{6.31}$$

Given Δt, a potential strategy involves estimating $F_{\max}$ and determining M to achieve the maximum desirable increment in velocity. Higher values along the diagonal of M correspond to reduced velocity increments.

Subsequently, the damping parameter is adjusted to achieve the desired time constants (Equation 6.24) and steady-state value (Equation 6.27), while adhering to the constraints outlined by Equation 6.28.

Block Diagram

A block diagram (Figure 6.5) provides a concise representation of all components. The forces and moments exerted by the human operator are measured by the sensor and reconstructed to the left and right handle forces, denoted as T_{hl} and T_{hr}. The handle forces, serve as input to the admittance Controller and the Patient Performance Selector (PPS).

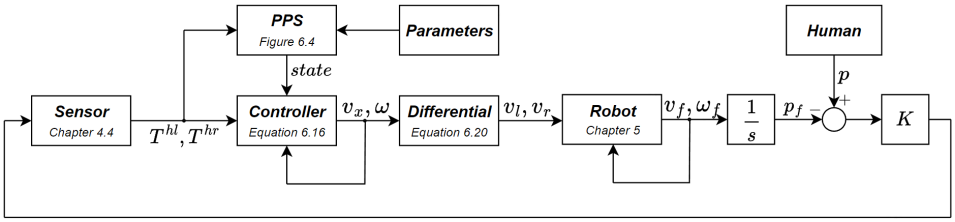


Figure 6.5: Block diagram of the overall control loop.

The PPS considers the needs and limitations of the patient by adjusting the force thresholds and the admittance parameters, denoted as Parameters. Additionally, the PPS updates its state based on the FSM outlined in Figure 6.4, where the state determines the Controller's action.

Furthermore, the Controller's output (Equation 6.14) is fed back for the next timestep. The output is further transformed into independent wheel velocities (v_l, v_r), as detailed in Equation 6.20, and applied to the robot. The resulting robot velocities (v_f, ω_f) are employed as feedback for the robot's inner control loop.

Additionally, the forward velocity v_f is integrated into position p_f and compared to the position of the human operator. According to Hooke's law, the resulting force is

$$F = K \cdot x, \tag{6.32}$$

where K is the human stiffness and $x = p - p_f$ is the displacement between the robot and the human. The force is fed back to the sensor, which closes the loop (Langeweg (2023)).

6.2 Fall Prevention

This section details the methodology employed to develop fall prevention strategies. It covers aspects such as monitoring weight distribution and detecting the support ellipse, as outlined in the theoretical framework of the thesis.

In line with the findings of Zhao et al. (2020), the vertical forces exerted by the user serve as indicators of weight distribution. Therefore, a significant difference between the vertical forces on the left and right handles may suggest a loss of balance. This is defined by

$$|f_{hlz} - f_{hrz}| > c_{\Delta z}, \tag{6.33}$$

where a difference exceeding the constant threshold $c_{\Delta z}$ is an indicator of emergency. To provide assistance, the robot will transition to the STOP state and cease its movement.

Moreover, an additional indication of an emergency is an excessively high vertical force, defined as

$$|f_{hlz} + f_{hrz}| > c_z \tag{6.34}$$

where $c_z >> c$, with c being the threshold from Equation 6.10. The two preceding strategies stop the robot's motion upon detecting an emergency, without adapting to offer additional support.

In contrast, the following approach adjusts the admittance parameters to provide assistance for users positioned too far from the support ellipse.

To discern between walking and emergency states, the laser range finder measures the user's position relative to the walker, denoted as $({}^{r}x_h, {}^{r}y_h)$. Moreover, the distance at which the laser range finder detects people is limited to an interval denoted as

$$\begin{aligned} {}^{r}x_h \in \quad & [x_{\min}, x_{\max}] \\ {}^{r}y_h \in \quad & [y_{\min}, y_{\max}]. \end{aligned} \tag{6.35}$$

Assuming the risk of falling occurs when the user is too far from the robot, the critical region is behind the user's average walking position. Thus, the minimum x-value is defined as $x_{\min} = \mu_x$.

The objective is to establish the maximum distance between the user and the support ellipse, which will inform the selection of the damping parameters. This involves defining d_e as the distance between the user and the edge of the support ellipse as follows

$$d_e = d_h - r_e = \sqrt{({}^{e}x_h - \mu_x)^2 + ({}^{e}y_h - \mu_y)^2} - \sqrt{\frac{1}{\left(\frac{\cos\theta_h}{a}\right)^2 + \left(\frac{\sin\theta_h}{b}\right)^2}}, \tag{6.36}$$

where θ_h is defined in Equation 4.32.

In order to determine the maximum value of d_e, it is necessary to identify the maximum value of d_h and then subtract the minimum value of r_e.

The maximum value of d_h is obtained when $({}^{e}x_h - \mu_x)^2 + ({}^{e}y_h - \mu_y)^2$ is maximized. This occurs when $({}^{e}x_h, {}^{e}y_h)$ is farthest from (μ_x, μ_y), which corresponds to $({}^{e}x_h, {}^{e}y_h)$ lying on the boundary of the specified range.

Similarly, the minimum value of r_e occurs when θ_h is maximized, corresponding to the point $({}^{e}x_h, {}^{e}y_h)$ being at the line $({}^{e}y_h = {}^{e}x_h)$. Consequently, the distance d_e achieves its maximum at x_{max}, y_{max} if $\mu_y < 0$, and at x_{max}, y_{min} if $\mu_y > 0$.

According to Equation 4.35, the modification of the damping parameters is a function of $d_e = d_h - r_e$. Consequently, a larger distance between the user and the support ellipse leads to higher damping parameters. This increase continues until the maximum distance $d_{e,\max}$ is obtained.

To achieve the desired damping parameters at the maximum distance, $d_{x,\max}$ and $d_{\theta,\max}$ must be defined, representing the maximum damping parameters allowed. These values are determined experimentally by identifying the stability limit of the system. Once all constants are established, the coefficients of the functions modifying the damping parameters are calculated as

$$\begin{aligned} c_x &= \frac{\ln(d_{x,\max} - d_{x,\text{init}})}{d_{e,\max}} \\ c_\theta &= \frac{\ln(d_{\theta,\max} - d_{\theta,\text{init}})}{d_{e,\max}}. \end{aligned} \tag{6.37}$$

6.3 Guidance

As defined initially, guidance involves the robot aiding and directing users along a predetermined route. The present study employs two distinct strategies to accomplish this. The first strategy involves the Variable Admittance Controller (Section 4.6.2), where users dictate motion while haptic feedback guides them along the correct route. The second solution utilizes the Kompaï robot's path planning algorithm, where an admittance controller governs forward linear velocity, while the robot's planner determines the optimal path to follow.

6.3.1 Variable Admittance Control

Guidance is achieved by dynamically modifying damping parameters according to user intention and deviation from the designated path. Consequently, the guidance framework builds upon the admittance controller utilized for walking assistance in 6.1.2.

To facilitate guidance, the first step involves mapping the robot's surroundings utilizing its laser scanner. Once a map is generated, waypoints are introduced to determine the path. This path comprises straight line segments connecting the designated waypoints. In cases where guidance is not initiated from the initial waypoint, the first line segment connects the robot's current position with the first designated waypoint.

Employing the LoS controller, the steering angle $\delta = \psi_{\text{los}}$ is determined by solving Equation 4.38 for x_{los} and y_{los}, followed by the application of Equation 4.37. Solving the system of equations in Equation 4.38 involves rewriting the second equation as

$$y_{\text{los}} = (x_{\text{los}} - x)\tan(\alpha_{k-1}) + y_{k-1}, \tag{6.38}$$

and substituting it into the first equation, resulting in

$$0 = ((x_{\text{los}} - x)\tan(\alpha_{k-1}) + y_{k-1} - y)^2 + (x_{\text{los}} - x)^2 - (nL)^2. \tag{6.39}$$

Moreover, Equation 6.39 represents a second-order equation, which implies that its solution can be expressed as

$$x_{\text{los}} = \frac{-b \pm \sqrt{b^2 - 4a}}{2a}. \tag{6.40}$$

Therefore, there exist two potential solutions for x_{los}, where the correct one is determined as the closest value to the current waypoint $p_k = \begin{bmatrix} x_k & y_k \end{bmatrix}^T$. Upon determining the correct x_{los} value, y_{los} is acquired by employing Equation 6.38.

After obtaining the LoS position, the steering angle δ is determined by applying Equation 4.37. Additionally, given that both the steering angle and heading angle are confined to the interval $[-\pi, \pi]$, it is required to maintain the steering error within the same range. Due to the angles wrapping around at 180 degrees, situations may arise where the heading angle and steering angle are in close proximity yet situated on opposite sides of the x-axis. Resulting in a difference close to 2π instead of zero. To ensure that the steering error $\tilde{\theta}$ is within the range of $-\pi$ to π, the following condition is applied

$$\tilde{\theta} = \begin{cases} \tilde{\theta} - 2\pi & : \tilde{\theta} > \pi \\ \tilde{\theta} + 2\pi & : \tilde{\theta} < \pi. \end{cases} \tag{6.41}$$

Equation 6.41 corrects for situations where the difference between the desired steering angle δ and the current orientation angle θ would exceed the specified interval. Thus, preventing large errors due to angle wrapping.

The varying admittance parameters are essential for achieving effective guidance as they provide the haptic sensation. To aid in the selection of damping parameters, the following relationship is derived from Equation 4.41

$$d_{x,\min} = d_{x,max} - d_{d,\max}, \tag{6.42}$$

as $0 < \exp(-x) \leq 1 \quad \forall \quad x \geq 0$.

As the steering error increases, it becomes more challenging to move the walker. Consequently, $d_{x,\min}$ should be selected to facilitate easy movement of the walker. Conversely, when experiencing a significant steering error, moving the walker becomes challenging. Therefore, $d_{x,max}$ is chosen as the highest possible damping value while adhering to the constraints specified in Equation 6.28.

Additionally, to aid in the selection of damping parameters for turning, the following relationships are derived from Equation 4.42

$$\begin{aligned} d_{\theta,\min} &= d_{i\theta} - G_{d\theta} \\ d_{\theta,\max} &= d_{i\theta} + G_{d\theta}, \end{aligned} \tag{6.43}$$

by utilizing the inequality $-1 < \tanh(x) < 1$. The selection of $d_{\theta,\min}$ and $d_{\theta,\max}$ follows similar principles to those discussed for d_x.

The designated coefficients are determined experimentally. Among these are δ_{dx} and $P_{d\theta}$, which influence the rate at which the damping parameters adapt to changes in error or user intention.

6.3.2 Guided Walk

The alternative approach to facilitate guidance, relies on the path planning algorithm of the Kompaï robot. This strategy utilizes the "Guided Walk" feature within the Kompaï software, accessible via the ROS interface of the robot. Guidance is achieved by iterating through all desired waypoints and directing the robot to reach each destination.

Additionally, the software enables dynamic adjustment of the linear forward velocity. Consequently, the robot's velocity along the path is regulated by user force through the implementation of an admittance controller in the forward direction.

6.4 Experimental Design

To evaluate the effectiveness of the developed solution, an experiment involving participants unfamiliar with the control system is conducted. This experiment aims to determine the intuitiveness and effectiveness of the interaction between humans and the robot. Furthermore, as the VAC for guidance provides haptic feedback to guide users along the correct path, participants being familiar with the route in advance would invalidate the experimental purpose. Therefore, executing the guidance prior to the walking assistance allows for the use of the same route without compromising the objectives of the guidance experiment.

6.4.1 Guidance

As previously stated, the aim of the guidance experiment is to evaluate the effectiveness of haptic feedback in providing cues for users to navigate along the path. The robot and the subject will be positioned at the starting point (Start) and informed of the location of the path's endpoint (End). To create an environment in which the user is dissuaded from attempting a direct route from the Start to the End, various obstacles are introduced to restrict the subject's movement. These obstacles are outlined in gray in Figure 6.6, alongside the waypoints represented as blue points.

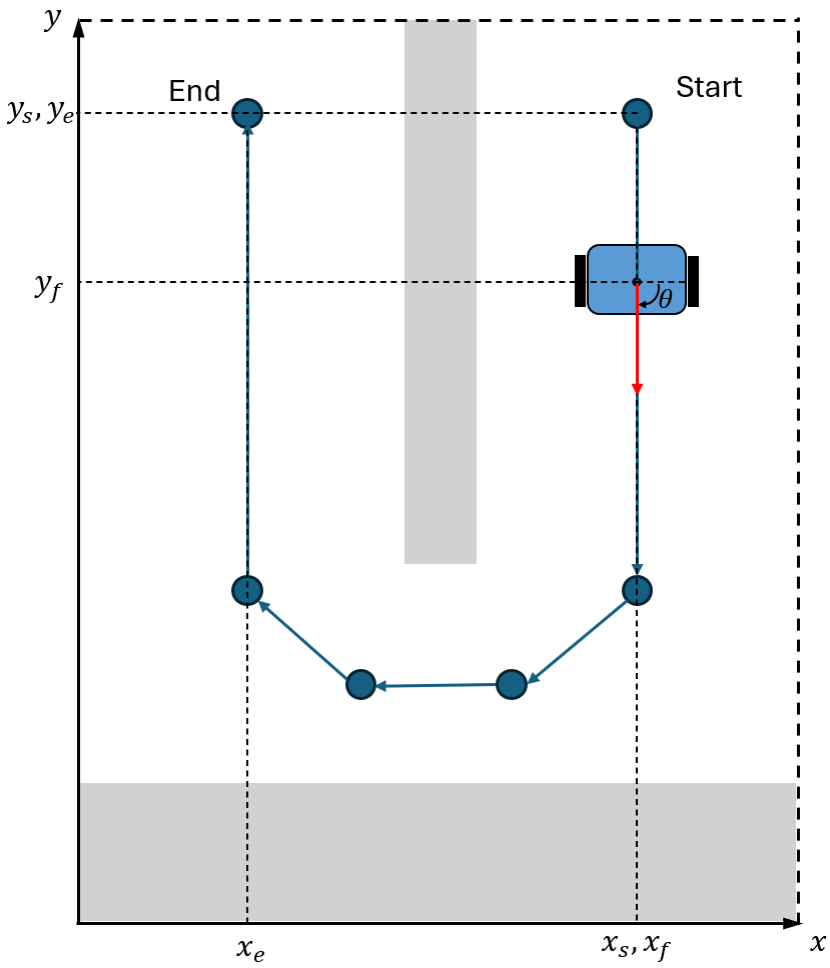


Figure 6.6: Waypoints outlining the experimental route.

However, the waypoints connecting the Start and End position will not be visible to the user. Consequently, any deviation from the path will depend on the effectiveness of the haptic feedback, as the user lacks visual cues regarding the correct direction to follow.

6.4.2 Walking Assistance

In the context of walking assistance, the aim is to evaluate the robot's maneuvering capabilities. This experiment will employ the identical route used for guidance. However, the waypoints and the pathway connecting them will be marked on the floor using masking tape. Therefore, the robot's maneuverability will be evident in deviations from the path caused by the forces exerted by the user.

6.4.3 Graphical User Interface

To optimize the experimental process for the participants and data recording, a Graphical User Interface (GUI) is developed utilizing FlexGui 4.0 (*FlexGui 4.0* (2022)), as illustrated in Figure 6.7.

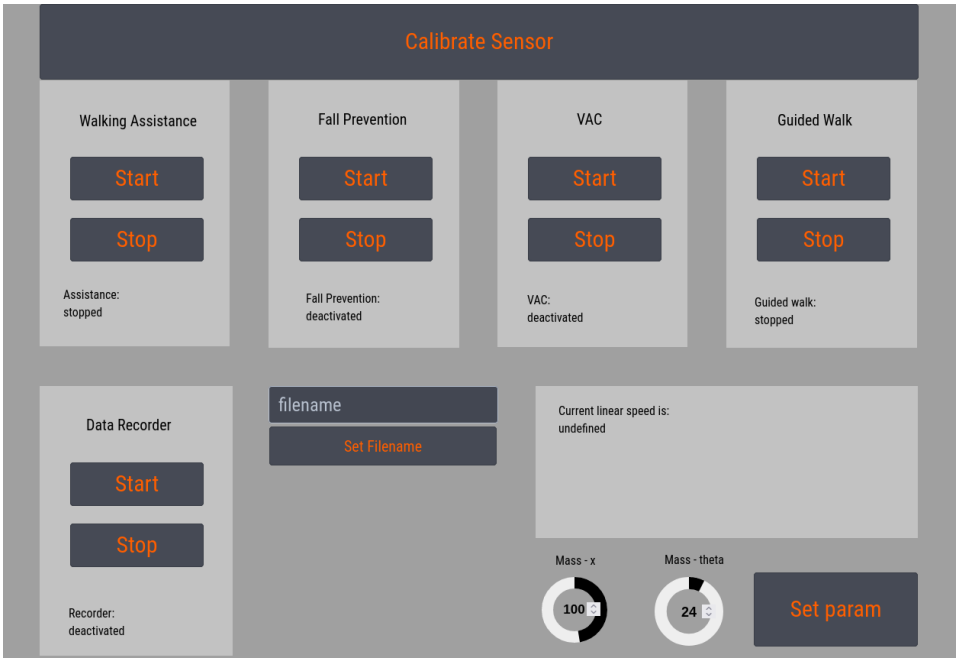


Figure 6.7: GUI implemented for startup.

The GUI consists of two mutually exclusive systems in Walking Assistance and Guided Walk. These functionalities employ separate interfaces for actuation and cannot be executed simultaneously. However, Fall Prevention and the VAC for providing Guidance are activated by dynamically adjusting the admittance parameters within the Walking Assistance's controller. Consequently, the activation of Guidance or Fall Prevention alongside Walking Assistance is necessary. These dependencies are clearly illustrated by a state machine presented in Figure 6.8, offering a comprehensive overview of the transitions between the different solutions.

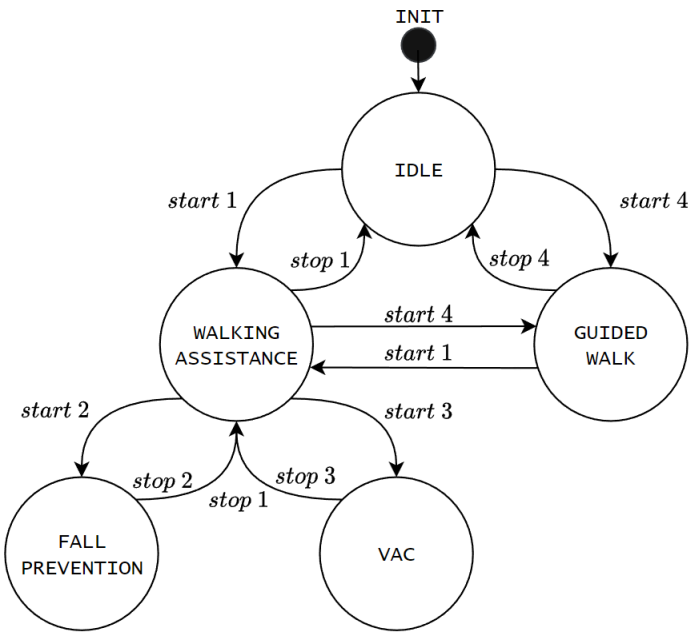


Figure 6.8: Transitions between the different solutions within the GUI.

In Figure 6.8, the enumeration of start and stop corresponds to the horizontal positioning of the widgets within the GUI (Figure 6.7), defined as

$$\begin{aligned} 1 &: \text{Walking Assistance} \\ 2 &: \text{Fall Prevention} \\ 3 &: \text{Guidance (VAC)} \\ 4 &: \text{Guided Walk.} \end{aligned} \tag{6.44}$$

Moreover, a solution for data recording and the corresponding metrics, as defined in Section 4.7, has been implemented under the title Data Recorder. This feature is initiated and terminated using buttons, consistent with the other functionalities. Additionally, within the GUI, users can specify the filename for the recording.

Furthermore, the mass parameters of the admittance controller can be modified using the “Set param” button. However, the parameter adjustments are not executed while the robot is running a program. Consequently, restarting the Walking Assistance is necessary for the updated parameters to be applied. Lastly, a button to update the force sensor offset has been included to mitigate the impact of sensor drift.

7

Experimental Results

In this chapter, the experimental results are presented, offering a detailed analysis of the implemented solutions. By examining the collected data, the study aims to draw meaningful conclusions regarding HRI with a robot tailored for the needs of the elderly.

7.1 Preliminary Results

The preliminary results address findings regarding the latency in the Kompaï robot's control system, as well as the sensor's noise, drift and the validation of handle force reconstruction. Additionally, a simulation of the admittance controller is conducted to measure the impact of the admittance parameters. These experiments aim to provide insights and validations for the subsequent implementation of the controller and the execution of upcoming experiments.

7.1.1 Step Response

The robot's response to a sudden step in commanded left and right wheel velocities is depicted in Figure 7.1. Three distinct approaches are employed, ranging from equal update and publish rates at 10Hz and 100Hz to a controller with different update (10Hz) and publish (100Hz) rates. The update rate denotes how frequently the reference signal is computed, while the publish rate indicates how often the reference signal is transmitted to the wheel motor drivers.

Moreover, the frequency of the measured velocity from the Kompaï robot is observed to be 20Hz. The latency originating from the Kompaï software imposes limitations on the update rate of the controller, making an update rate exceeding 20Hz ineffective in enhancing performance. However, employing a publish rate higher than the update rate is observed to yield the lowest time constant (τ). The time constant is determined as the duration until the speed reaches $\frac{1}{e} = 63.2\%$ of the reference (Weik (2001)), resulting in $\tau_1 = 0.18$, $\tau_2 = 0.12$, and $\tau_3 = 0.09$ for the three distinct approaches in Figure 7.1.

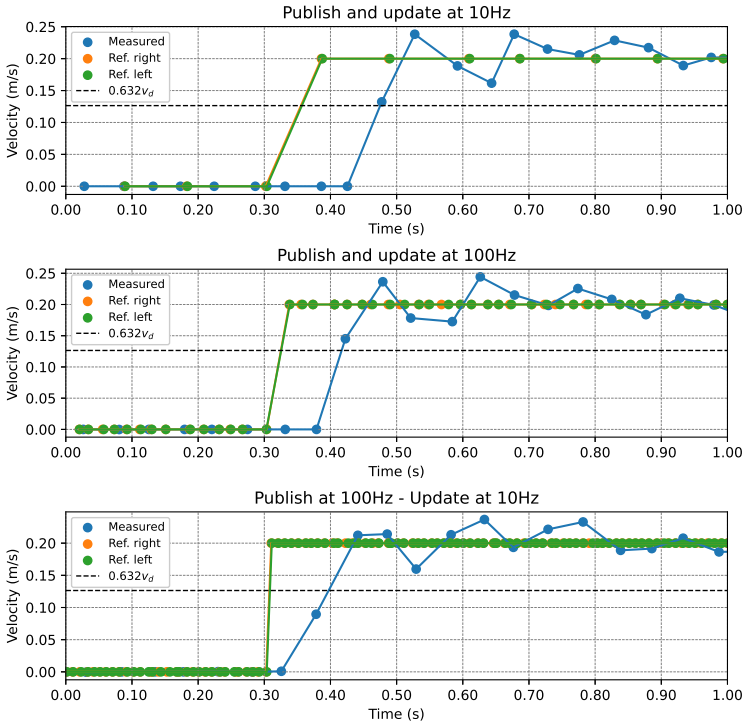


Figure 7.1: Step response in velocity.

7.1.2 Force Sensor

The figure below illustrates the drift in the z-axis of the force sensor over a span of 19 hours in the absence of external forces. It demonstrates a noticeable shift in the z-component of the force measurements from its initial values.

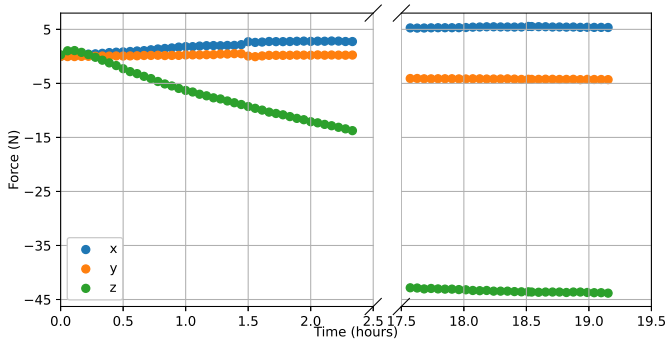


Figure 7.2: Drift in sensor data.

Furthermore, the sensor exhibits a certain level of noise, particularly evident in the z-axis with a magnitude of 0.15N (Langeweg (2023)).

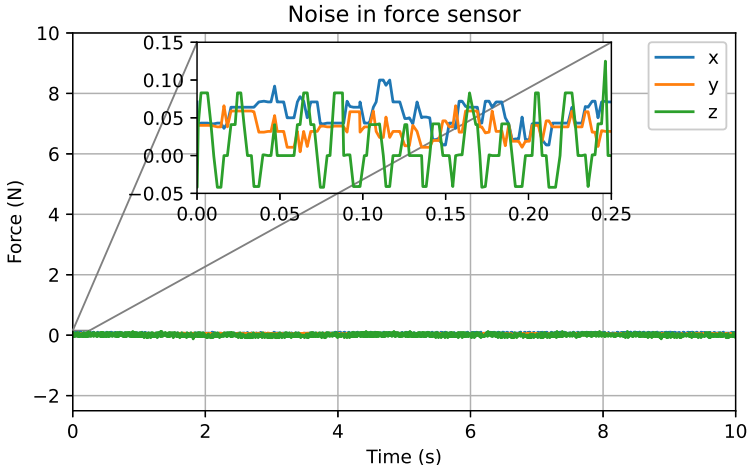


Figure 7.3: Noise in sensor data taken from Langeweg (2023).

The following figure illustrates the validation of handle force reconstruction from measured forces/moments in the sensor frame T_s, as conducted in Langeweg (2023). At $t = 2$s, a force is exerted on the left handle. The figure illustrates a nearly perfect alignment between the left handle force F_{hlx} (orange) and the measured force F_{sx} (blue), with the right handle force F_{hrx} (green) showing a slight deviation from zero. At $t = 10$s, the experiment is replicated with a force applied solely to the right handle. This results in F_{hrx} closely tracking the F_{sx}, while the F_{hlx} displays a marginal deviation from zero. These results confirm that the reconstruction method accurately computes the handle forces.

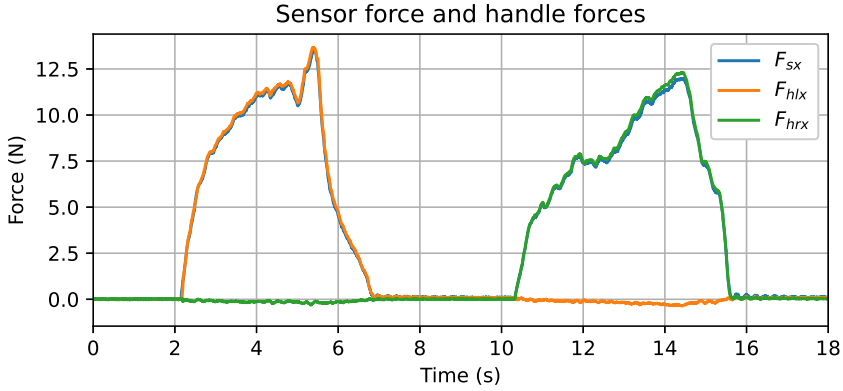


Figure 7.4: Measured forces vs handle forces, generated by Langeweg (2023).

7.1.3 Simulation of Admittance Controller

Using the derived time-dependent output response of the admittance controller (Equation 6.22), the admittance model is simulated in Figure 7.10 to observe the impact of different admittance parameters.

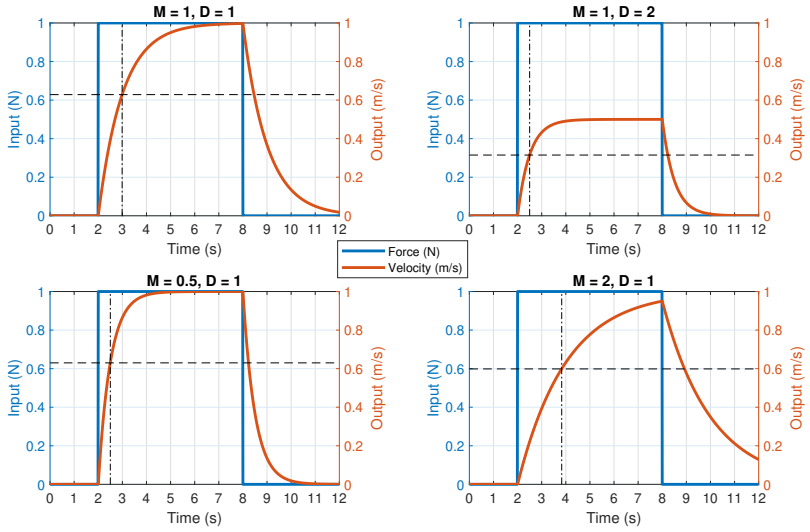


Figure 7.5: Simulation of admittance model with various admittance parameters.

Using the parameters outlined in Figure 7.10, the time constants are calculated following Equation 6.24. Additionally, a step response in the applied force f is simulated for a duration of 6 seconds. With a constant force of $f^* = 1$N, the steady-state velocity is determined using Equation 6.27. These findings are summarized in Table 7.1.

Table 7.1: Admittance parameters, time constants and steady state velocity.

$\mathbf{M}$	$\mathbf{D}$	$\mathbf{T_c}$	$\mathbf{v}^*$
1	1	1	1
1	2	0.5	0.5
0.5	1	0.5	1
2	1	2	1

Moreover, as indicated by Equation 6.25, it is evident that the position of the admittance model diverges while the velocity converges to zero. To validate this theory, an additional simulation is conducted where both position and velocity are displayed, as illustrated in Figure 7.6.

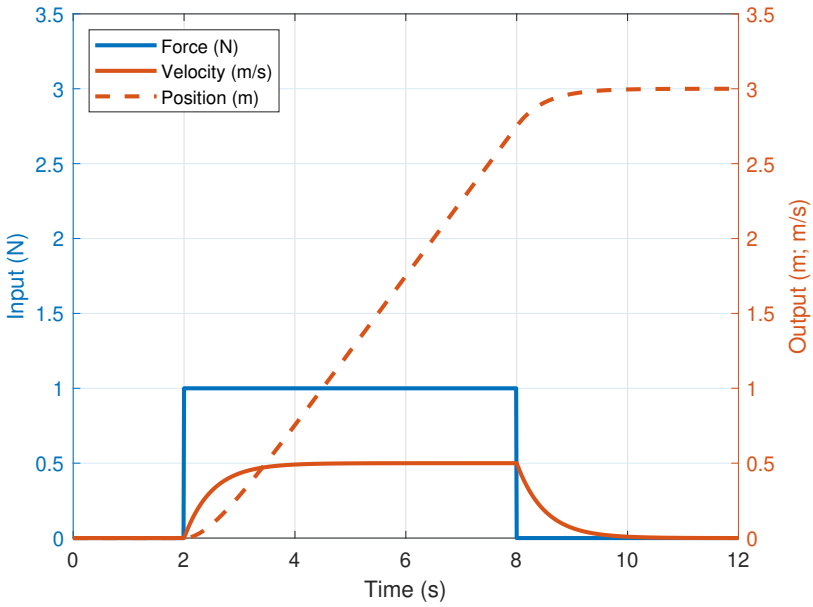


Figure 7.6: Simulated system response to a step in input force.

7.2 Walking support

Regarding walking support, the experiments investigate the various turning strategies utilized in conjunction with the controller developed in the project report Langeweg (2023), alongside a comparison with the developed admittance controller. Additionally, experiments are conducted to quantify the impact of various admittance parameters on robot behavior and interaction dynamics.

7.2.1 Turning Strategies

The following experiment aims to assess and compare the different turning strategies, based on the controller introduced in Langeweg (2023), with the admittance controller developed in this thesis. These results are obtained by applying identical force profiles to all approaches for a predefined duration.

As depicted in Figure 7.7, the force in the y-direction experiences a step to the minimum force needed to initiate a turn, then rises for 5 seconds, followed by a decrease of equal duration before dropping to zero. While F_{sy} remains consistent across all approaches, F_{sx} differs between the admittance controller and the turning strategies. The turning strategies rely on an impulse in F_{sx} to initiate velocity incrementation, as it builds on the controller from Langeweg (2023).

In contrast, the admittance controller requires a non-zero force to generate velocity. Hence, F_{sx} is maintained at a constant force of 15N for the admittance controller. This

explains why the admittance controller shown in Figure 7.7 commands a forward velocity differing from that of the various turning strategies.

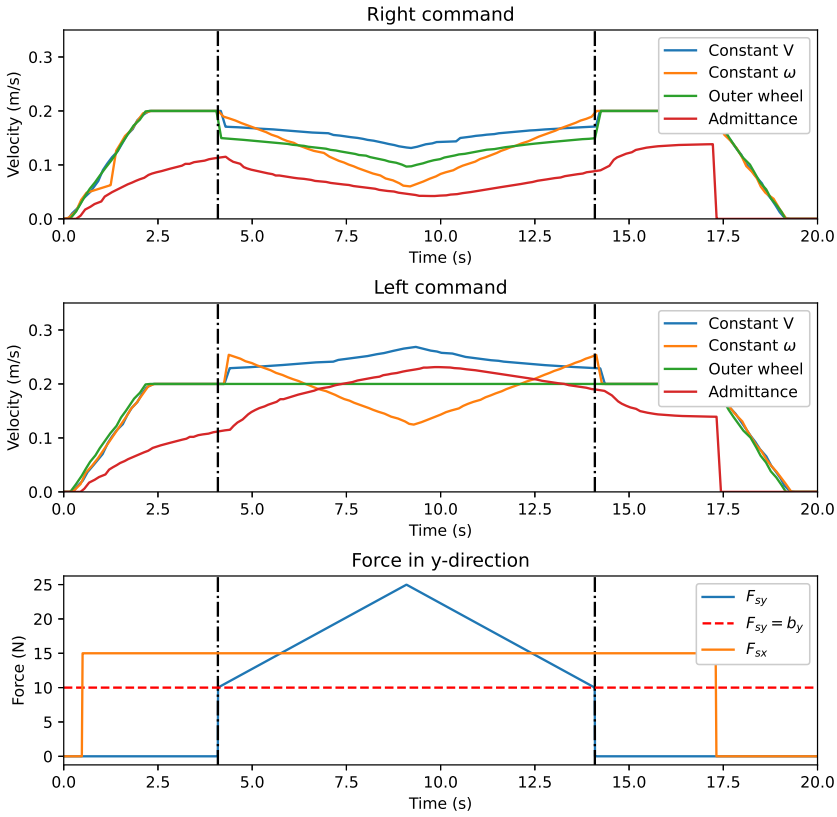


Figure 7.7: Comparison of actuation.

To evaluate the turning strategies detailed in Section 6.1.1, denoted as constant v, constant ω and constant outer wheel velocity, the parameters specified in Equation 6.3 must be assigned. These parameters involve the minimum and maximum force F_y required to scale the turning radius R. The minimum turning radius is given by $R_{\min} = \frac{L}{2} + \text{margin}$, where L denotes the length of the axle connecting the two wheels. Additionally, a constant angular velocity ω_d is chosen for the constant ω strategy.

The experimental parameters are assigned as follows:

$$\begin{array}{lcll} F_{y,\min} & = & 10.0 & [\text{N}] \\ F_{y,\max} & = & 30.0 & [\text{N}] \\ R_{\max} & = & 1.5 & [\text{m}] \\ L & = & 0.43 & [\text{m}] \\ \text{margin} & = & 0.1 & [\text{m}] \\ \omega_d & = & 0.15 & [\text{rad/s}]. \end{array} \tag{7.1}$$

The velocity commands assigned by the various controllers (Figure 7.7) result in the following positions in space.

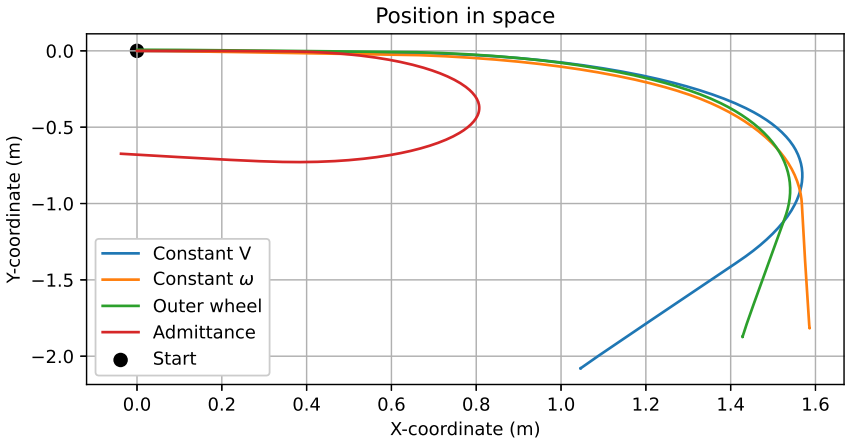


Figure 7.8: Position in space obtained with equal force.

Furthermore, the extent to which the different approaches turn is investigated by plotting their respective heading angles over time. As depicted in Figure 7.9, the admittance controller completes a turn of more than 180 degrees in 10 seconds, whereas the other methods achieve turns closer to 90 degrees over the same duration.

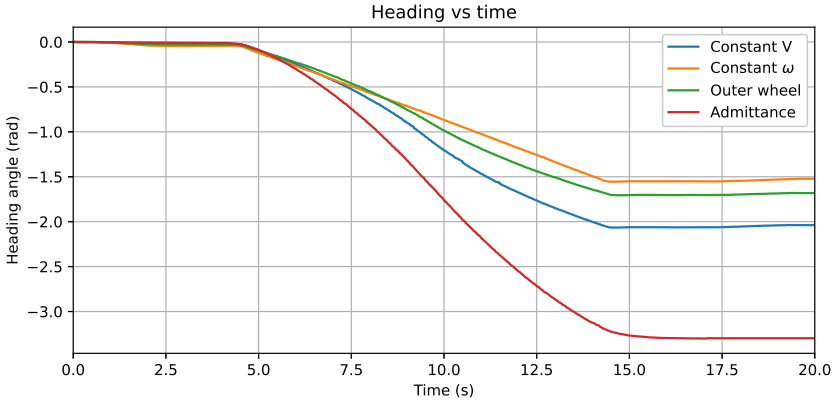


Figure 7.9: Comparison of heading angle.

7.2.2 Admittance Controller

Admittance Parameters

In order to gain insights into the impact of the admittance parameters on the walking assistance, a series of experiments are conducted. By keeping one of the admittance parameters constant while varying the other, the effect of either damping or mass can be isolated. In the following two experiments, a force equal to that of the experiment in Figure 7.7 is applied. For the first one, the damping parameters are kept constant while the mass parameters are varied according to Equation 7.2.

$$\mathbf{M}_I = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}; \quad \mathbf{M}_{II} = \begin{bmatrix} 50 & 0 \\ 0 & 50 \end{bmatrix}; \quad \mathbf{M}_{III} = \begin{bmatrix} 250 & 0 \\ 0 & 250 \end{bmatrix} \tag{7.2a}$$

$$\mathbf{D} = \begin{bmatrix} 100 & 0 \\ 0 & 50 \end{bmatrix} \tag{7.2b}$$

The results obtained from varying the mass parameters are displayed in Figure 7.10. It is observed that by increasing the mass parameter, the time constant increases, which is in accordance with the relation derived in Equation 6.24.

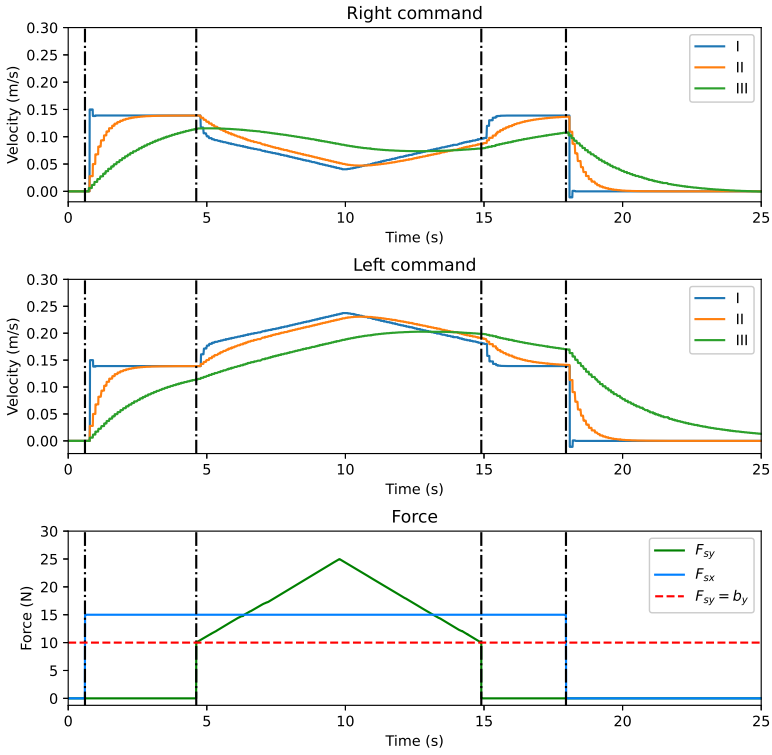


Figure 7.10: Comparison of different mass parameters.

For the second experiment, the mass parameters are kept constant while the damping parameters are varied according to Equation 7.3.

$$\mathbf{D}_I = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}; \quad \mathbf{D}_{II} = \begin{bmatrix} 50 & 0 \\ 0 & 50 \end{bmatrix}; \quad \mathbf{D}_{III} = \begin{bmatrix} 250 & 0 \\ 0 & 250 \end{bmatrix} \tag{7.3a}$$

$$\mathbf{M} = \begin{bmatrix} 100 & 0 \\ 0 & 50 \end{bmatrix} \tag{7.3b}$$

By increasing the damping parameters in Figure 7.11, it is evident that the same amount of force yields a lower steady state velocity. Furthermore, higher damping values lead to a reduction in the time constant. These observations are consistent with the relationships described in Equation 6.27 and Equation 6.24.

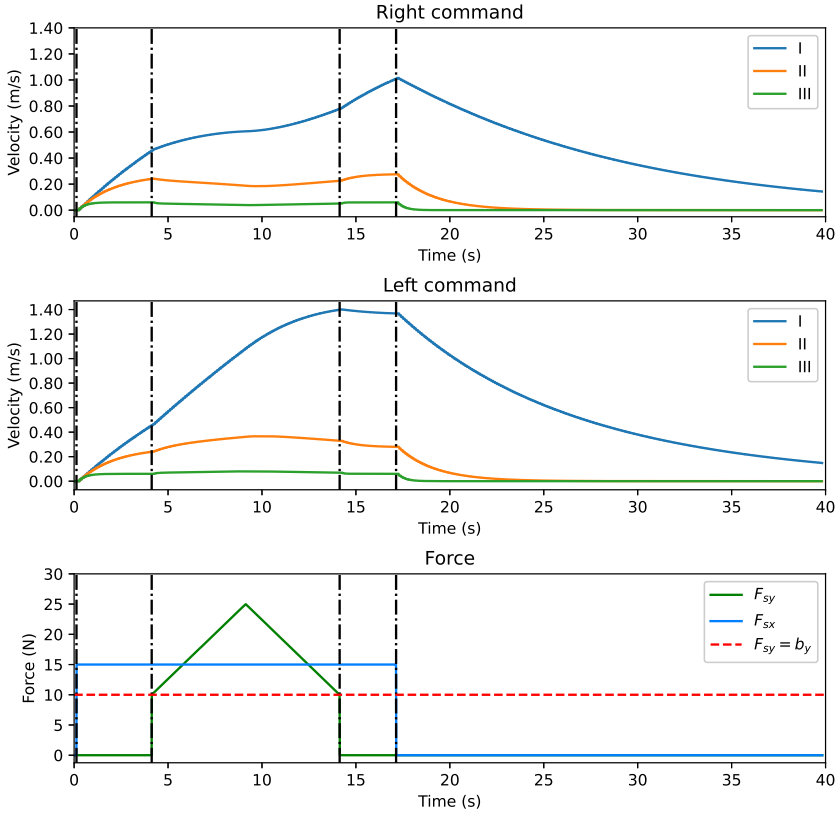


Figure 7.11: Comparison of different damping parameters.

Update Rate

An experiment resembling that shown in Figure 7.1 is carried out with the admittance controller to investigate the influence of the controller's update frequency. At 0.2 seconds into each trial, a constant force is applied. As depicted in Figure 7.12, three distinct approaches are employed. The first two maintain identical update and publish rates at 10Hz and 100Hz, respectively, while the third utilizes a 10Hz update rate and a 100Hz publish rate.

Throughout this particular experiment, the admittance parameters remain unchanged, resulting in a different change per unit force for the controller with a 100Hz update rate, as detailed in Equation 6.31. Consequently, the results indicate that the middle controller with 100Hz, equivalent to $\Delta t = 0.01$, converges more rapidly towards the steady-state value.

Additionally, it is apparent that the final controller, characterized by different update and publish rates, repeats the same reference ten times over. This recurrence is facilitated by the publication rate being ten times higher than the update rate. Therefore, this strategy results in the smoothest measured velocity.

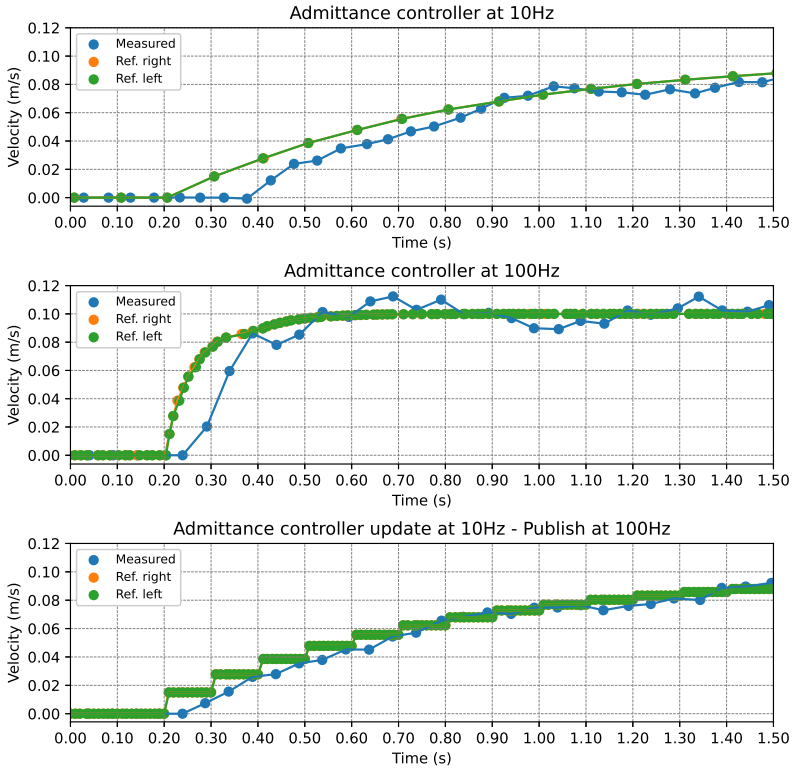


Figure 7.12: Comparison of different update and publish rates for the admittance controller.

To justify the selection of the controller's update rate, an additional experiment is conducted. The update and publish rates replicate those utilized in the previous experiment involving a step in the applied force. However, in this scenario, a user aims to sustain a constant force in the x-direction for a duration of 10 seconds. Additionally, ensuring a similar contribution per unit force at various update rates requires the relationship outlined in Equation 6.31 to be consistent across the various approaches. Therefore, for an update rate at 10Hz, equivalent to $\Delta t = 0.1$, the following parameters are chosen

$$\mathbf{M} = \begin{bmatrix} 100 & 0 \\ 0 & 24 \end{bmatrix}; \mathbf{D} = \begin{bmatrix} 150 & 0 \\ 0 & 36 \end{bmatrix}, \tag{7.4}$$

while for 100Hz, resulting in a timestep at $\Delta t = 0.01$, the parameters are

$$\mathbf{M} = \begin{bmatrix} 10 & 0 \\ 0 & 2.4 \end{bmatrix}; \mathbf{D} = \begin{bmatrix} 150 & 0 \\ 0 & 36 \end{bmatrix}. \tag{7.5}$$

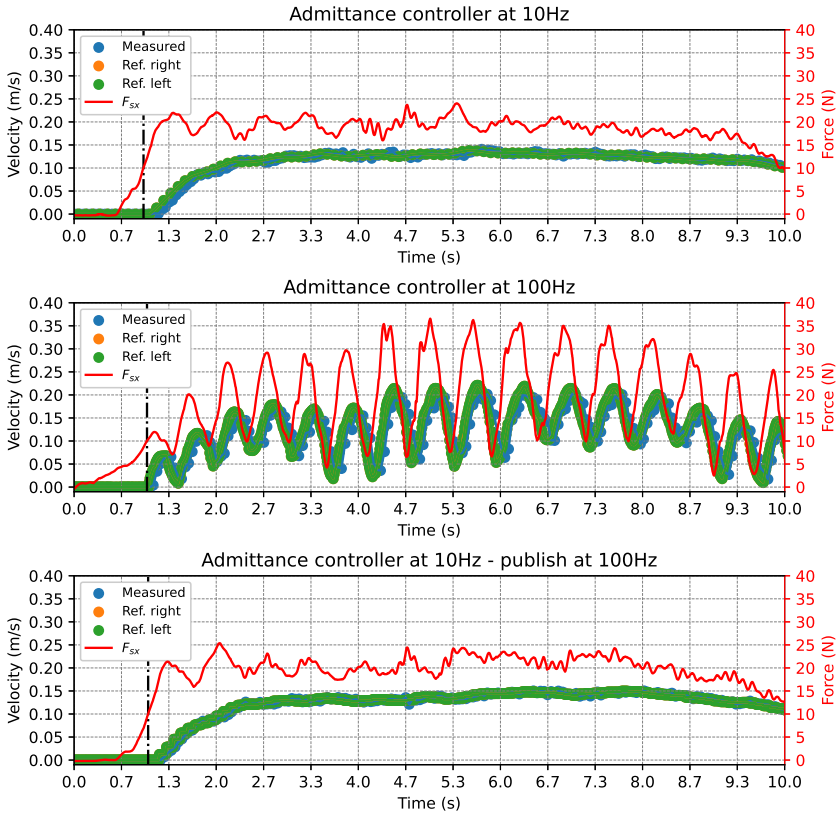


Figure 7.13: Comparison of different update and publish rates with user applied force.

Given these parameters, the relation from Equation 6.31 is obtained as follows

$$\Delta t = 0.1: \quad m_x = 100, m_\theta = 24 \quad \rightarrow \quad \frac{(d_x \cdot \Delta t)}{m_x} = \frac{(d_\theta \cdot \Delta t)}{m_\theta} = 0.15$$
$$\Delta t = 0.01: \quad m_x = 10, m_\theta = 2.4 \quad \rightarrow \quad \frac{(d_x \cdot \Delta t)}{m_x} = \frac{(d_\theta \cdot \Delta t)}{m_\theta} = 0.15,$$

confirming that one unit force contributes equally to the velocity increase across all controllers.

By examining Figure 7.13, it is evident that the interaction between the user and the robot becomes oscillatory at an update rate of 100Hz. This occurs despite the user's efforts to maintain a constant force level. The admittance controller commands velocities that exhibit oscillations, leading to oscillatory forces applied by the user as they hold onto the handles. Further investigation into the potential reasons behind this behavior will be addressed in the discussion.

The following figure offers a closer view of the preceding experiment depicted in Figure 7.13. The admittance controller is configured to initiate movement for $F_{sx} \geq 10$N, as a means to mitigate oscillations attributed to noise in the force sensor.

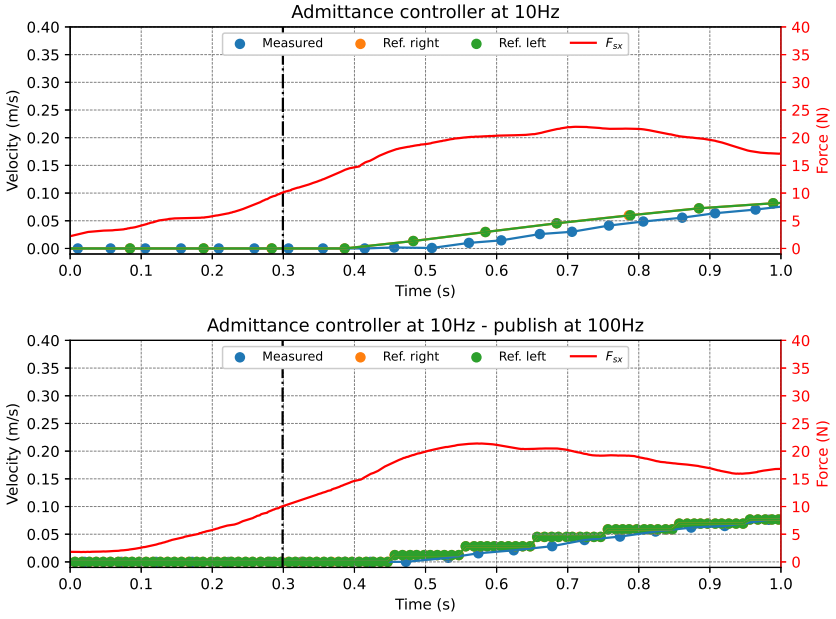


Figure 7.14: Closer view of the initial second in Figure 7.13.

Moreover, the vertical dashed line in Figure 7.14 denotes the point at which the force exceeds the threshold. Initially, it appears that there is a delay between the force surpassing the threshold and the reference velocity increasing.

In addition, the data points along the reference serve to indicate the controller's update frequency. This observation reveals that there is only a single time step between the detection of the force level and the subsequent increase in reference velocity. This phenomenon is less noticeable in the second subplot, where the reference is published at 100Hz. Nonetheless, since the update rate remains consistent with the previous scenario, the same explanation applies.

To illustrate the effect of downsampling the force signal, the following simulation is conducted. A combination of sinusoidal waves is employed to replicate the interaction forces between the user and the handles. Initially, a slowly varying sinusoidal with an amplitude of 5N and an offset of 20N is introduced to emulate the gait phase of the walker. Subsequently, a fast varying sinusoidal of 2N is incorporated to represent the high-frequency variations inherent in the continuous interaction between the user and the robot. As the user walks, postural sway and minor oscillations from the robot's chassis induce high-frequency variations in the interaction force. Furthermore, to account for sensor noise, an additional high-frequency sinusoidal with a small amplitude is included.

As demonstrated in Figure 7.15, sampling the force signal at 100Hz preserves numerous high-frequency variations. Conversely, sampling at 10Hz effectively captures the overall trend of the force signal. Although the noise is not eliminated, the user intention is obtained.

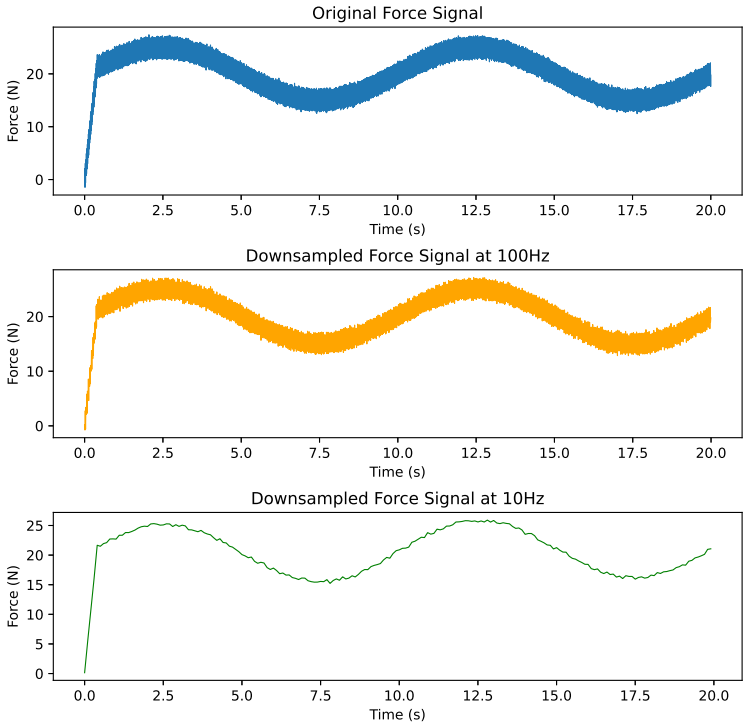


Figure 7.15: Different sampling rates of the force signal.

Stability Limit

In order to achieve a comprehensive understanding of the admittance controller, the following experiment is conducted. It involves the execution of a step in applied force with various damping parameters.

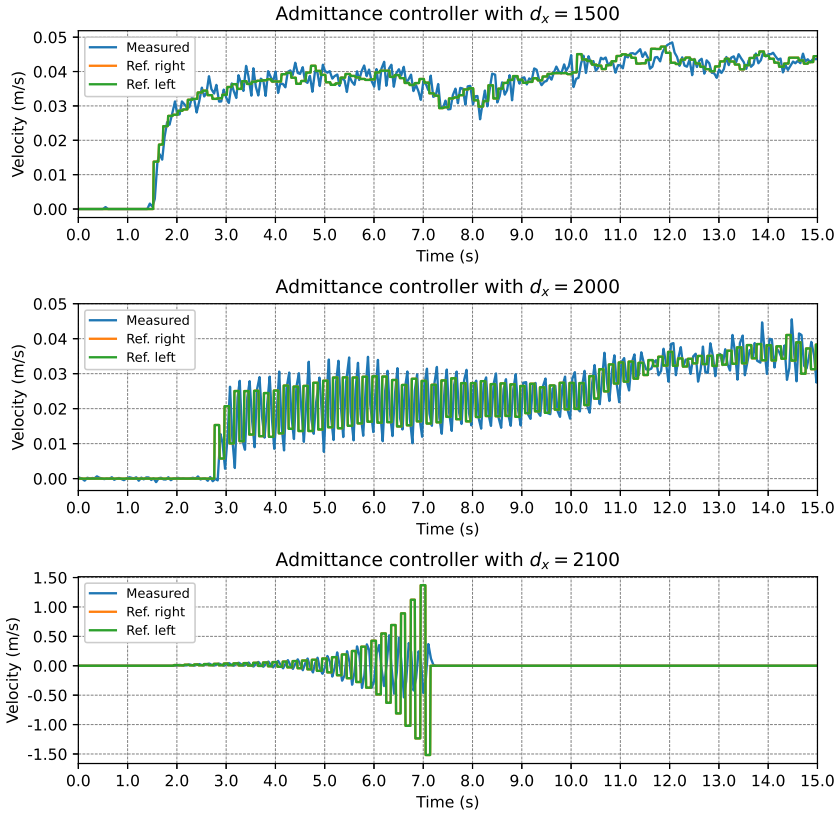


Figure 7.16: Analysis of the stability limit for the admittance controller.

Given an update rate of $\Delta t = 0.1$ and a mass parameter of $m_x = 100$, the damping parameters depicted in Figure 7.16 yield the following values when applying the relation from Equation 6.28:

$$\Delta t = 0.1: \quad m_x = 100, d_x = 1500 \quad \rightarrow \quad \frac{(d_x \cdot \Delta t)}{m_x} = 1.5$$

$$\Delta t = 0.1: \quad m_x = 100, d_x = 2000 \quad \rightarrow \quad \frac{(d_x \cdot \Delta t)}{m_x} = 2.0$$

$$\Delta t = 0.1: \quad m_x = 100, d_x = 2100 \quad \rightarrow \quad \frac{(d_x \cdot \Delta t)}{m_x} = 2.1,$$

where $\frac{(d_x \cdot \Delta t)}{m_x} > 1$ results in an oscillatory admittance model. However, according to Equation 6.29, the admittance model becomes unstable only when $\frac{(d_x \cdot \Delta t)}{m_x} > 2$. Upon examining the behavior of systems with different damping, it becomes apparent that these relations remain valid.

Cost Evaluation

In the subsequent figure, an attempt is made to evaluate the impact of different damping parameters. The objective is to walk a distance of 2 meters with the robot while observing the costs associated with an increase in the damping parameter.

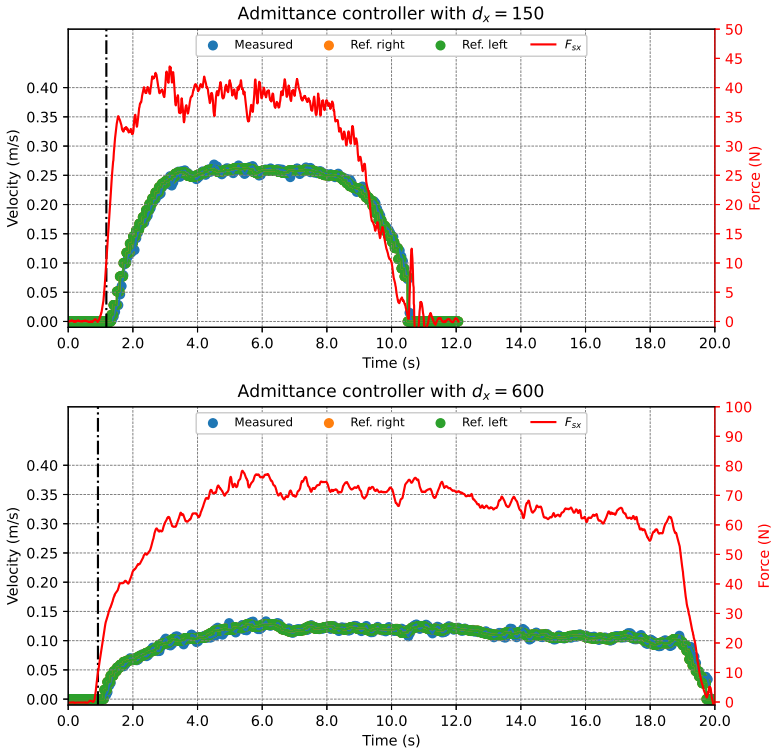


Figure 7.17: Walk of 2m for admittance controllers with different damping d_x.

The performance metrics outlined in Section 4.7 are utilized to assess these costs and summarized in Table 7.2.

The cost related to the higher damping parameter ($d_x = 600$) becomes evident in several aspects. Firstly, the completion time nearly doubles, compared to that of the lower damping parameter ($d_x = 150$). Moreover, both the work performed by the user and the impulse transmitted to the robot notably increase. This distinction is evident in Figure 7.17, where the applied force is higher for the higher damping, while the commanded velocity

Table 7.2: Cost evaluation for admittance parameters with different d_x values.

Parameter	$d_x = 150$	$d_x = 600$
Distance [m]	2.00	1.98
Duration [s]	12.08	20.87
Impulse [Ns]	313.97	1191.16
Work [Nm]	52.11	113.14
Correlation	0.89	0.41

is lower. Lastly, the correlation between the walker speed and the speed of the user serves as a metric for assessing the controller's performance, revealing a significantly lower correlation for the high damping parameter.

7.3 Fall prevention

In this section, experiments are conducted to evaluate fall prevention strategies, integrating the support ellipse to detect emergency situations. These experiments seek to reveal valuable insights and provide information on determining coefficients for adjusting admittance parameters, thereby offering additional support for the user.

7.3.1 Support Ellipse

To establish the axes of the support ellipse, the mean position (μ_x, μ_y) of the user behind the Kompaï robot is recorded, along with the computation of the standard deviations σ_x and σ_y. Applying $\alpha = 0.1$ in Equation 4.27 to establish a 90% confidence interval, the support ellipse is depicted alongside the recorded walker position in Figure 7.18.

Additionally, to generalize across users and environments, the correlation coefficient ρ is set to zero, as discussed in the subsequent chapter. The descriptive statistics are summarized in Equation 7.10, with measurements provided in millimeters. This includes the magnitude of the semi-major a and semi-minor b axes for the specific trial utilized in determining the support ellipse.

$$\begin{aligned} \mu_x &= 430.3 \ [\text{mm}] \\ \mu_y &= 2.2 \ [\text{mm}] \\ \sigma_x &= 51.7 \ [\text{mm}] \\ \sigma_y &= 38.8 \ [\text{mm}] \\ a &= 110.9 \ [\text{mm}] \\ b &= 83.2 \ [\text{mm}] \end{aligned} \tag{7.6}$$

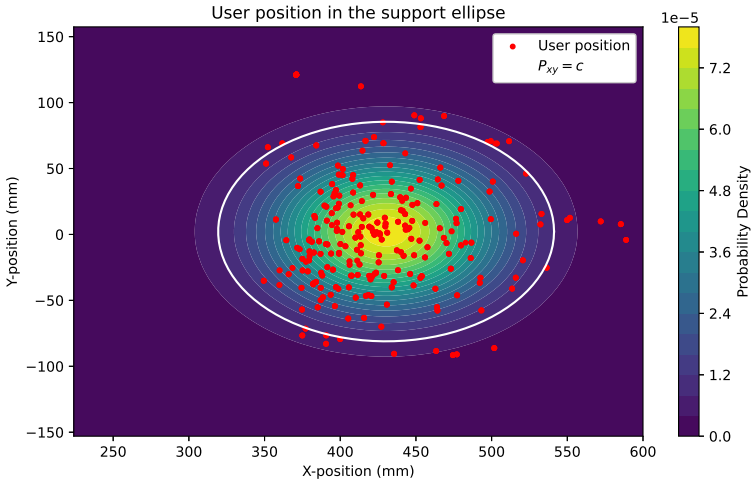


Figure 7.18: User position in the support ellipse.

In Figure 7.19, a red point denotes the position representing the maximum distance from the walker to the user.

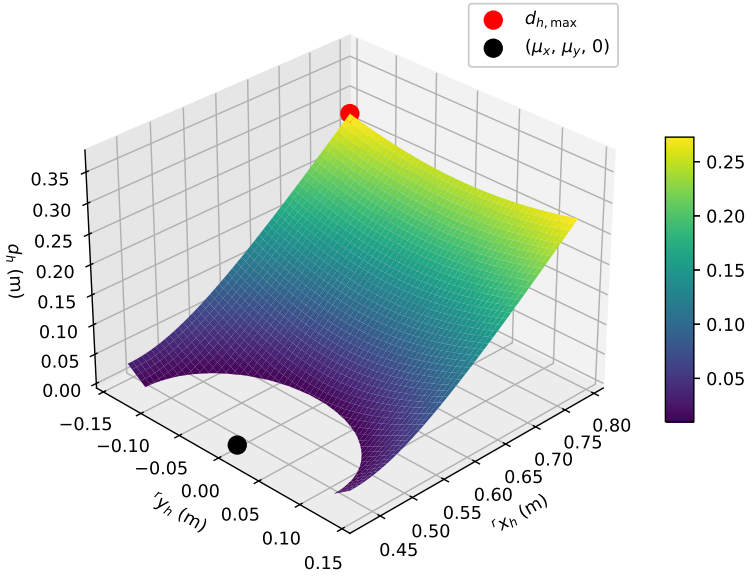


Figure 7.19: Maximum distance from the walker robot to the user.

By analyzing the data from the laser scanner, the maximum range for user detection, in meters, is determined within the intervals

$$\begin{aligned}[x_{\min}, x_{\max}] &= [0, 0.79] \\ [y_{\min}, y_{\max}] &= [-0.139, 0.139].\end{aligned} \tag{7.7}$$

Additionally, as derived in Section 6.2, the distance d_h between the walker and the user reaches its maximum at x_{max}, y_{min} when $\mu_y > 0$. With $\mu_y = 0.0022$ [m], the maximum value is achieved at the coordinate $(0.79, -0.139)$, where the maximum distance is $d_{h,\max} = 0.277$ meters.

7.3.2 Parameters

The function governing the adjustment of admittance parameters (Equation 4.35) determines the behavior of the fall prevention strategy. In normal operational conditions, the admittance parameters are determined as follows

$$\mathbf{M}_{\text{init}} = \begin{bmatrix} 100 & 0 \\ 0 & 24 \end{bmatrix}; \mathbf{D}_{\text{init}} = \begin{bmatrix} 150 & 0 \\ 0 & 36 \end{bmatrix}, \tag{7.8}$$

where these parameters enable the determination of the relationship described in Equation 6.28, resulting in

$$\begin{aligned}\Delta t = 0.1: \quad m_x = 100, d_x = 150 \quad &\rightarrow \quad \frac{(d_x \cdot \Delta t)}{m_x} = 0.15 \\ \Delta t = 0.1: \quad m_\theta = 24, d_x = 36 \quad &\rightarrow \quad \frac{(d_\theta \cdot \Delta t)}{m_\theta} = 0.15.\end{aligned}$$

To determine the maximum allowable parameters $d_{x,\max}$ and $d_{\theta,\max}$, the stability criteria outlined in Section 6.1.2 are employed. According to Equation 6.28, a relation between the mass and damping less than 1 prevents oscillations. Nevertheless, to provide additional support, a certain degree of oscillation is acceptable. Thus, the relation should not exceed 2, as specified by the stability limit in Equation 6.29. Through trial and error, the maximum damping parameters are established as $d_{x,\max} = 1900$ and $d_{\theta,\max} = 400$, resulting in

$$\begin{aligned}\Delta t = 0.1: \quad m_x = 100, d_{x,\max} = 1900 \quad &\rightarrow \quad \frac{(d_x \cdot \Delta t)}{m_x} = 1.90 \\ \Delta t = 0.1: \quad m_\theta = 24, d_{\theta,\max} = 400 \quad &\rightarrow \quad \frac{(d_\theta \cdot \Delta t)}{m_\theta} = 1.67.\end{aligned}$$

With the previously defined parameters as well as the maximum distance between the user and the walker $d_{e,\max}$, the coefficients in Equation 4.35 are derived using Equation 6.37, yielding

$$\begin{aligned}c_x &= \frac{\ln(d_{x,\max} - d_{x,\text{init}})}{d_{e,\max}} = 27.2 \\ c_\theta &= \frac{\ln(d_{\theta,\max} - d_{\theta,\text{init}})}{d_{e,\max}} = 21.3.\end{aligned} \tag{7.9}$$

7.3.3 Evaluation

The performance of the fall prevention system is evaluated with the established coefficients for adaption of admittance parameters in an emergency state.

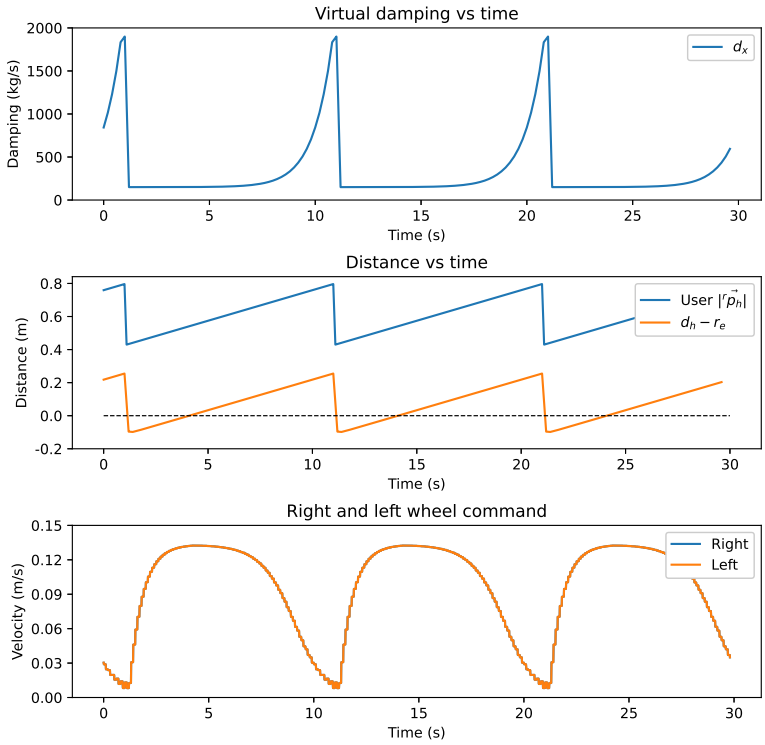


Figure 7.20: Adjustment of damping parameter d_x for fall prevention.

In Figure 7.20, a constant force of $F_{sx} = 20\text{N}$ is exerted. Meanwhile, the user's position relative to the walker robot, represented by $|^r\vec{p}_h| = \sqrt{{}^r x_h^2 + {}^r y_h^2}$, linearly increases within the interval $[\mu_x, \sqrt{x_{\max}^2 + y_{\min}^2}] = [0.43, 0.80]$. The user's distance is confined within the specified interval, resulting in a periodic behavior resembling a sawtooth function.

As depicted in Figure 7.20, once the distance from the user to the walking ellipse ($d_e = d_h - r_e$) exceeds zero, there is a corresponding increase in the damping parameter d_x. With this increase in the damping parameter, the velocity gradually decreases, despite the application of a constant force. Moreover, the presence of oscillations in the commanded velocity becomes evident as the damping parameter approaches its maximum value, a point elaborated in the discussion.

Similarly, in Figure 7.21, a constant force of $F_{sy} = 20\text{N}$ is applied alongside the same variation in the distance to the user, as observed in the previous experiment. With a positive force applied, the user intends to turn left, causing the left wheel to move forward and the

right wheel to move backward. As seen in Figure 7.20, when $d_e = d_h - r_e$ exceeds zero, it leads to an increase in the damping parameter d_θ. Consequently, the velocity is reduced to near zero. Although the robot does not come to a complete stop, the velocity is notably reduced to provide increased support.

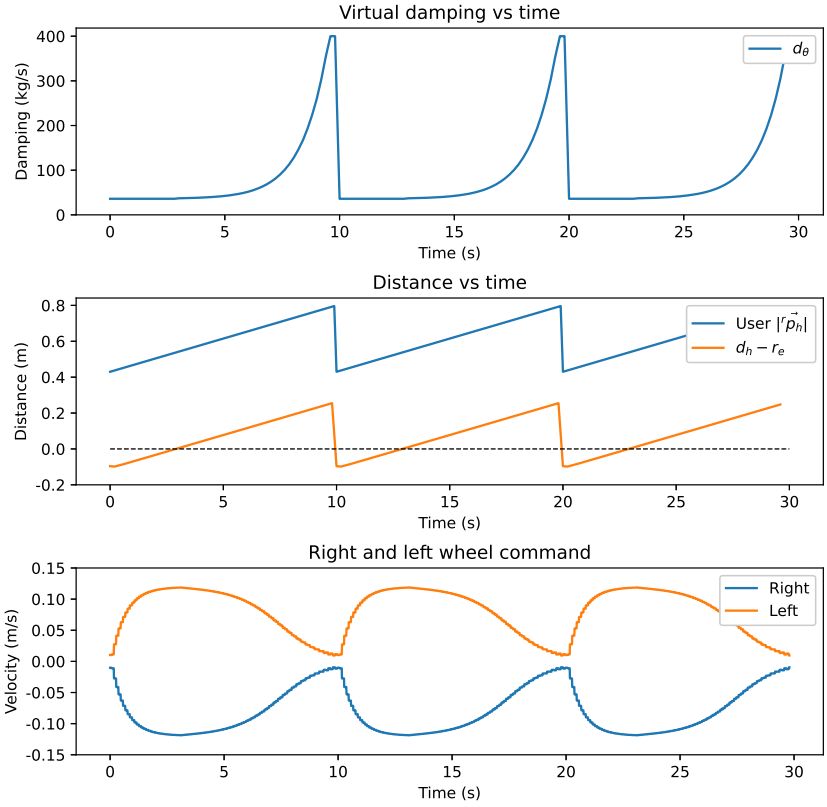


Figure 7.21: Adjustment of damping parameter d_θ for fall prevention.

7.4 Guidance

Within this section, experiments are conducted to provide information on the coefficients for the VAC to facilitate guidance. Furthermore, the two guidance strategies are evaluated and compared.

7.4.1 Parameters

As guidance is facilitated by the adaptation of admittance parameters, the selection of coefficients governing this adjustment is crucial to ensure efficient guidance.

The damping parameter for linear velocity is a function of the steering error, as detailed in Equation 4.41. Following a similar reasoning as for the fall prevention parameters, the maximum allowable damping parameter is set to $d_{x,\max} = 1900$. Furthermore, the minimum damping parameter is defined by the initial damping parameters for non-emergency conditions, as defined in Equation 7.8, resulting in $d_{x,\min} = 150$.

Additionally, the mass parameter m_x is chosen to be identical to the value used for fall prevention in Equation 7.8, namely $m_x = 100$. Employing the relationship outlined in Equation 6.42, the coefficients are then determined as follows

$$\begin{aligned} d_{x,\max} &= 1900 \\ d_{d,\max} &= 1750. \end{aligned} \tag{7.10}$$

Furthermore, the coefficient δ_{dx}, dictating how fast the damping parameter adjusts in relation to the steering error, is determined through experimentation. Figure 7.22 demonstrates the influence of δ_{dx}, assisting in the selection process. To provide users with some autonomy and facilitate smoother transitions to new waypoints, δ_{dx} is chosen as $\delta_{dx} = 1.4$.

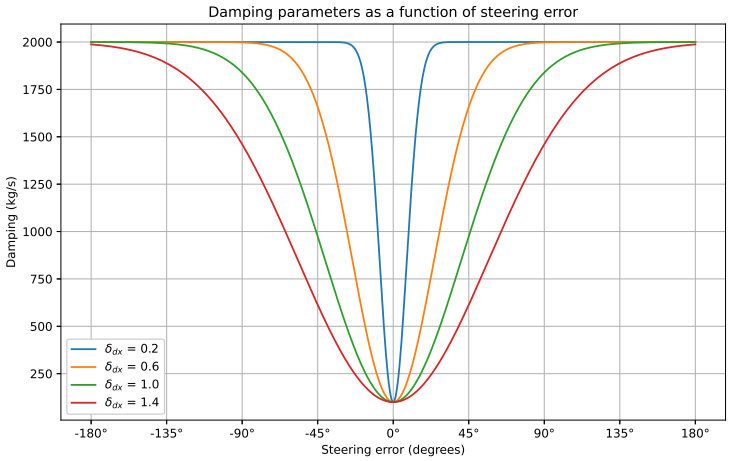


Figure 7.22: Impact of δ_{dx} on the damping parameter d_x for guidance.

In contrast to the damping affecting linear movement, the damping parameter d_θ for turning (Equation 4.42) is influenced by both the steering error and the applied force in the y-direction. Furthermore, the maximum acceptable value for this parameter does not align with that of fall prevention. Instead, it is chosen as the highest feasible value to mitigate oscillations, as ensured by Equation 6.28. Nevertheless, the minimum value and the constant mass remain identical to the parameters defined for fall prevention under non-emergency conditions, as outlined in Equation 7.8. Thus, the mass parameter is defined as $m_\theta = 24$ and the minimum damping parameter is set to $d_{\theta,\min} = 36$.

By applying Equation 6.28 to the mass parameter with $\Delta t = 0.1$, the maximum damping parameter is determined as

$$\frac{d_{\theta,\max} \cdot \Delta t}{m_\theta} = 1 \quad \rightarrow \quad d_{\theta,\max} = \frac{m_\theta}{\Delta t} = 240. \tag{7.11}$$

Having determined $d_{\theta,\min}$ and $d_{\theta,\max}$, Equation 6.43 is rewritten as

$$\begin{aligned} G_{d\theta} &= \frac{d_{\theta,\max} - d_{\theta,\min}}{2} = 102 \\ d_{i\theta} &= d_{\theta,\max} - G_{d\theta} = 138, \end{aligned} \tag{7.12}$$

to determine the coefficients governing the adjustment of the damping parameter d_θ.

Furthermore, the coefficient $P_{d\theta}$, that governs the rate at which the damping parameters adjust to changes in error or user intention, is established through experimentation. Given its dependency on both steering error and applied force, Figure 7.23 is divided into two figures, each representing a constant force level with different signs applied. It becomes evident that higher values of $P_{d\theta}$ lead to slower transitions to increased damping. The selection of this coefficient entails a trade-off between accuracy and autonomy. In subsequent experiments, $P_{d\theta}$ is set to $P_{d\theta} = 1.6$.

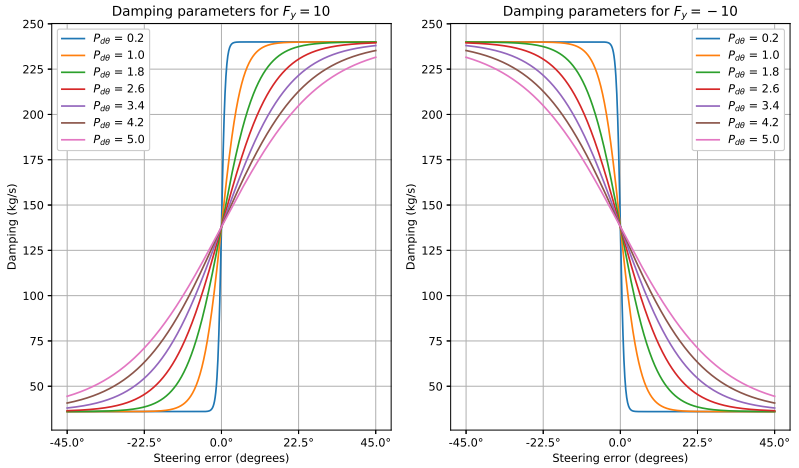


Figure 7.23: Impact of $P_{d\theta}$ on the damping parameter d_θ for guidance.

Additionally, for a deeper understanding of the influence of steering error and applied force on the damping parameter d_θ, Figure 7.24 is provided. The strategy to achieve the lowest damping parameter when the steering error $\tilde{\theta}$ is greater than zero involves applying a negative force in the y-direction (F_{sy}) and vice versa.

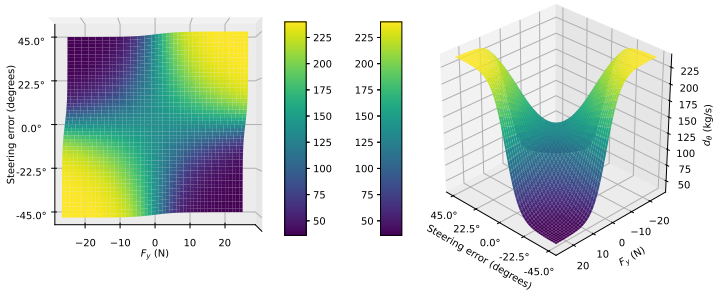


Figure 7.24: Impact of steering error and applied force on the damping parameter d_θ.

7.4.2 Evaluation

Upon determining the coefficients for adjusting admittance parameters to facilitate guidance, the performance of the VAC is assessed.

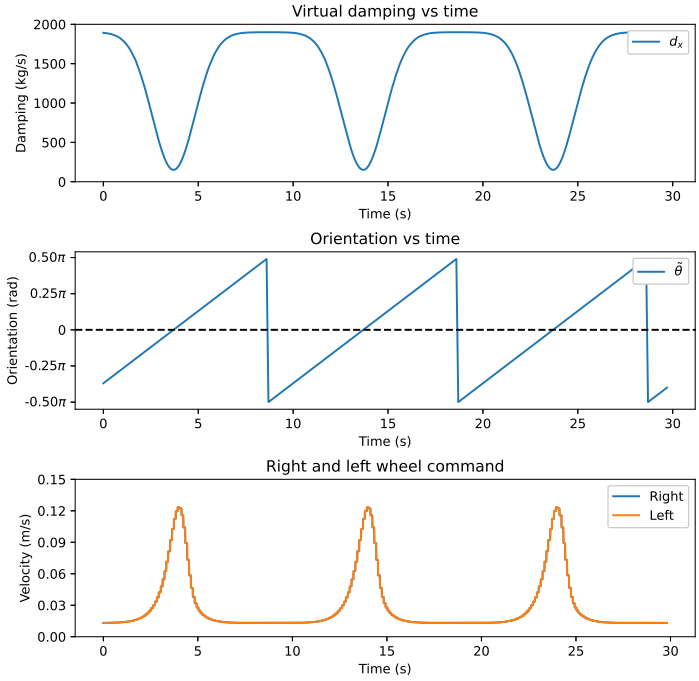


Figure 7.25: Adjustment of damping parameter d_x for guidance.

In Figure 7.25, a constant force of $F_{sx} = 20$N is applied. Simultaneously, the steering error $\tilde{\theta}$ increases linearly within the interval $[-\frac{\pi}{2}, \frac{\pi}{2}]$. The steering error is confined within this specified range, resulting in periodic behavior resembling a sawtooth function. As illustrated in Figure 7.25, when the steering error is near zero, the damping parameter d_x remains low, leading to high velocity. However, as the absolute value of the error increases, the damping parameter increases accordingly, causing the velocity to decrease. Consequently, the user is restrained from moving in the incorrect direction.

Conversely, in Figure 7.26, the applied force F_{sy} increases linearly within the interval $[-20, 20]$ with periodic behavior. Additionally, a constant steering error of $\tilde{\theta} = -\frac{\pi}{4}$ is simulated. From Figure 7.26, it becomes apparent that a positive force combined with a negative steering error corresponds to the user's intention of correcting the error. Consequently, the velocities of the left and right wheels increase with positive and negative signs, respectively, turning the robot to the right to minimize the error. However, applying a negative force causes the wheel velocities to approach zero, preventing the error from escalating.

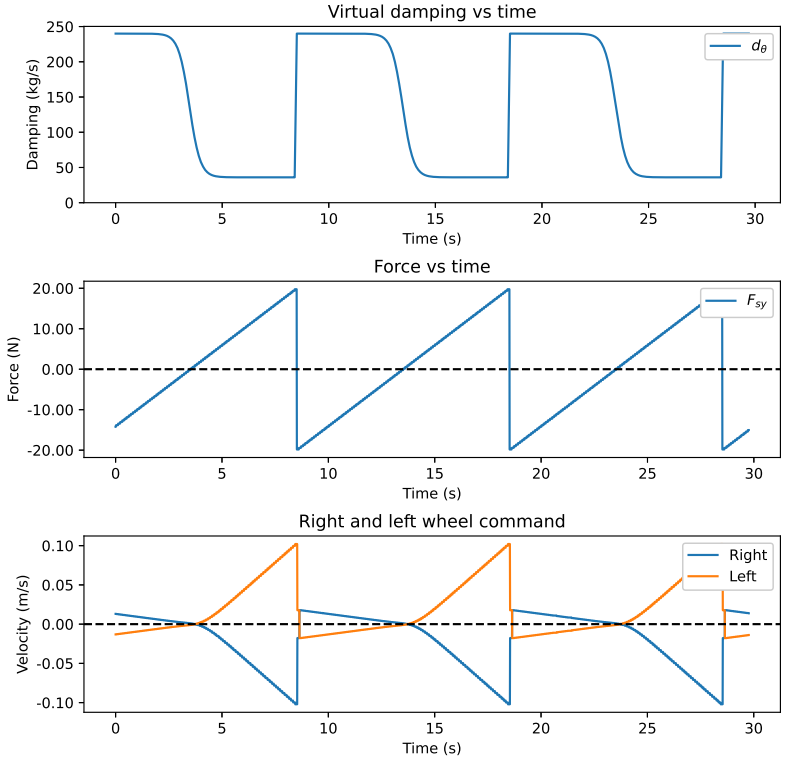


Figure 7.26: Adjustment of damping parameter d_θ for guidance.

Additionally, in Figure 7.27, an experiment is conducted with a maximum damping parameter exceeding the value suggested by Equation 6.28. As a result, oscillations in

the wheel velocities become apparent during the transition between low and high damping parameters.

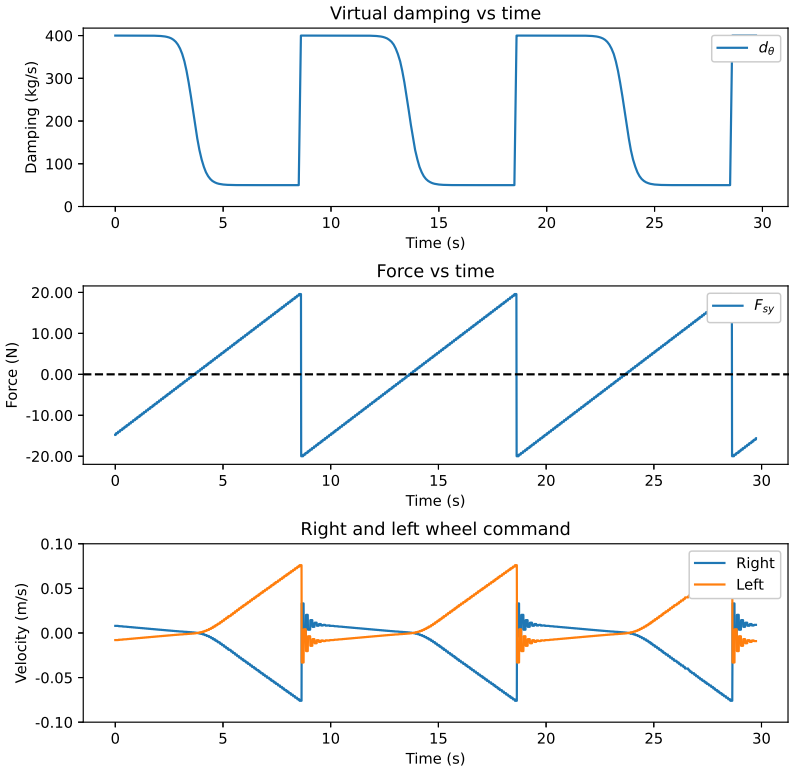


Figure 7.27: Oscillation in wheel velocities with too high damping parameters.

7.4.3 VAC vs Guided Walk

To highlight the contrast between the guidance solutions enabled by the VAC and the guided walk facilitated by Kompaï's path planner, the same experiment is conducted using both approaches. The objective is to follow a predefined set of waypoints as indicated in Figure 7.28.

From Figure 7.28, it is apparent that the trajectory followed with VAC closely resembles a series of straight line segments connecting the waypoints. Conversely, the path taken with the guided walk appears smoother but deviates from the path. Specifically, it passes directly through waypoint W1 but overshoots waypoint W2. The blue transparent circles represent the radius in which a waypoint is considered reached, prompting the transition to the next waypoint. Hence, it is essential to consider the impact of the acceptance radius, particularly for closely spaced waypoints.

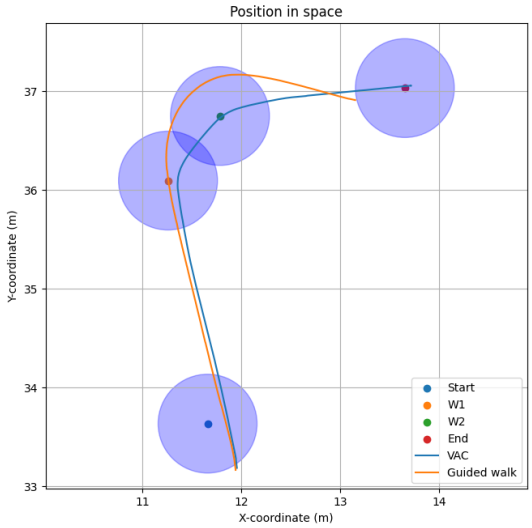


Figure 7.28: Comparison between VAC and guided walk by traversing a set of waypoints.

Additionally, presenting the forces and forward linear velocities for both the walker robot and the user, enables a comparison between the two systems. The forward linear velocity of the robot during VAC, as shown in the bottom left of Figure 7.29, confirms the previously discussed observation of this solution being less smooth. In contrast, the guided walk's velocity profile displays a noticeably smoother pattern. Moreover, the smooth transition in the robotic walker's speed contributes to a better synchronization with the human user's velocity.

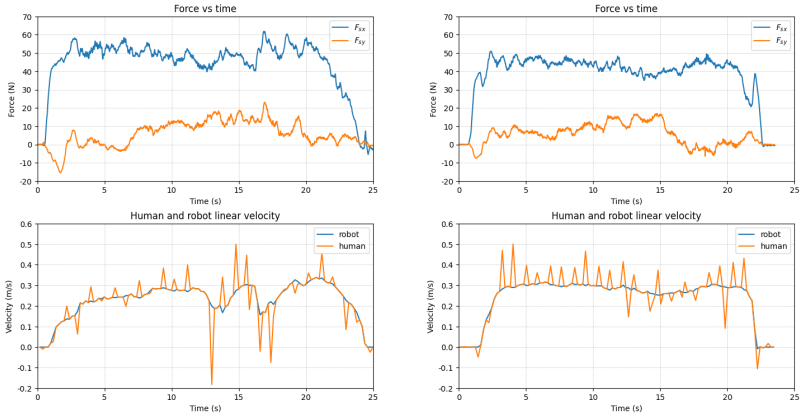


Figure 7.29: Comparison of applied forces and forward linear velocity for VAC (left) and guided walking (right).

Moreover, the metrics outlined in Section 4.7 are captured to assess performance and are summarized in Table 7.3. These metrics reveal that the completion time and distance traveled are nearly identical for both approaches. However, the correlation validates the observation made in Figure 7.29, regarding the better synchronization between the user's and the walker's speed during the guided walk. Furthermore, the impulse and work exerted during the guided walk are lower compared to VAC, as evident in the aforementioned figure. Lastly, the cross-track error (XTE) is notably higher for the guided walk, as observed in Figure 7.28.

Table 7.3: Comparison of performance metrics for Guided Walk and VAC.

Parameter	Guided Walk	VAC
Distance [m]	5.71	5.73
Duration [m]	23.54	25.64
Impulse [Ns]	921.14	1125.17
Work [Nm]	223.45	251.34
Correlation	0.91	0.84
XTE	322.20	12.61

7.5 Experiments with Human Subjects

Within this section, experiments involving five participants are conducted. The subjects have limited familiarity and knowledge of the solution. The objective, as outlined in Section 6.4, is to assess the effectiveness of the VAC for guidance and the admittance controller for walking support by navigating a predefined route.

7.5.1 Guidance

In the guidance experiment, participants are only provided with information about the starting and ending points, as detailed in Section 6.4. Each participant completes three trials, with the initial and final trials depicted in Figure 7.30. The blue transparent circles in the figure represent the acceptance radius of the waypoints, indicating when a waypoint is considered reached. The trajectories depicted reveal that all participants showed improvement from the first trial, as they became acquainted with the system. Notably, subject 4 demonstrates proficient performance from the first trial, a topic explored in the discussion.

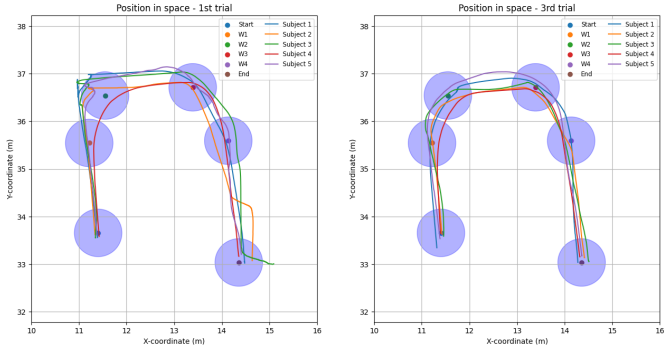


Figure 7.30: A comparison of the path traversed during the first and third trials.

Furthermore, Figure 7.31 depicts the spatial position over time, with the time dimension represented along the z-axis. This visualization facilitates a comparison of the paths traversed by the subjects and the time taken to accomplish the task. Additionally, a gradient is applied to the trajectory to illustrate the steering error at each location. The figure reveals that subjects encounter challenges in reducing steering error, particularly near the waypoints where turning is required.

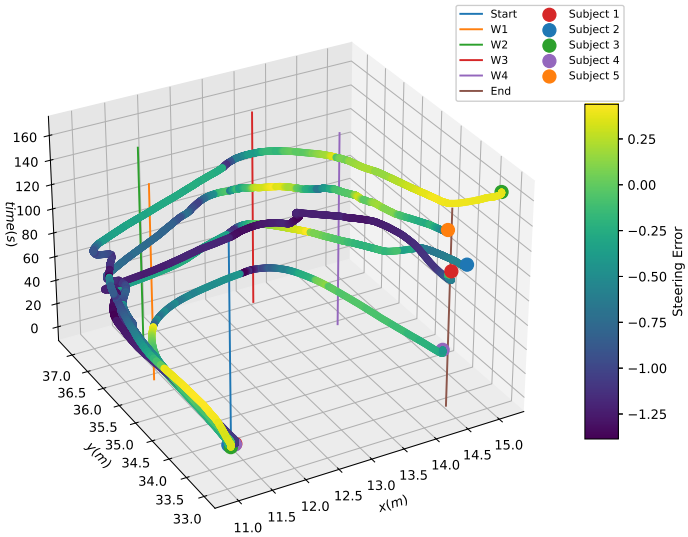


Figure 7.31: Comparing the trajectories and steering errors across the subjects' first trials.

Additionally, the subsequent figure presents the same plot as described above, however, for the third trial of each subject. As trajectories become smoother, an improved performance is noticeable. Steering errors decrease more rapidly after transitioning to new waypoints and completion times are generally shorter. However, whether this enhancement is due to the user's familiarity with the path or increased adaptation to the system, will be explored in the discussion.

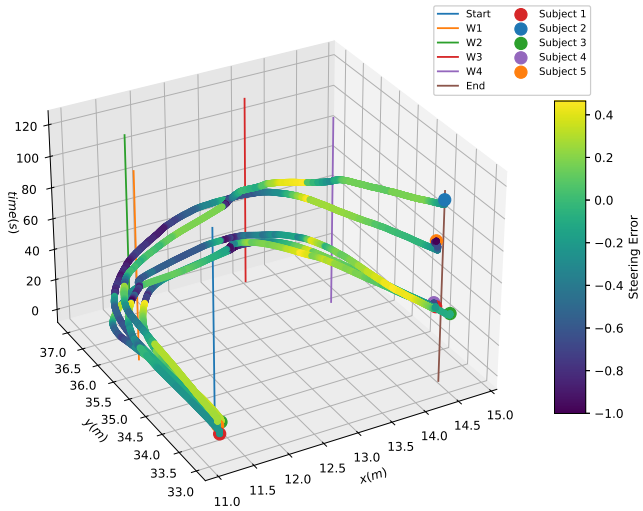


Figure 7.32: Comparing the trajectories and steering errors across the subjects' third trials.

The results of the experiments are evaluated using the previously mentioned metrics, with data from the first and third trials presented in Table 7.4 and Table 7.5, respectively. Notably, improvements in the third trial include reductions in total distance and completion time, as well as a decrease in impulse. However, variations in performance among subjects are observed in the correlation between walker and user speed, work and XTE, topics that will be addressed later.

Table 7.4: Performance metrics recorded for the subjects during the first trial.

Metric	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5
Correlation	0.29	0.23	0.15	0.54	0.16
Distance [m]	10.8	9.8	11.3	8.6	10.0
Duration [s]	107.9	110.7	163.5	43.0	138.1
Impulse [Ns]	4434.7	2337.3	3185.7	1658.2	3185.6
Work [Nm]	445.8	206.7	220.0	333.3	230.6
XTE [m]	37.9	1223.8	2470.4	3482.8	28.6

Table 7.5: Performance metrics recorded for the subjects during the third trial.

Metric	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5
Correlation	0.17	0.05	0.23	0.42	0.40
Distance [m]	9.4	8.6	9.1	8.3	8.5
Duration [s]	48.9	112.1	41.4	48.7	55.1
Impulse [Ns]	1847.2	1703.1	1872.5	1604.5	1605.9
Work [Nm]	356.3	130.1	409.7	273.1	248.0
XTE [m]	94.7	1245.5	2479.6	3493.8	49.1

From these experiments, two distinct user behaviors emerge. In general, participants tend to increase their speed as they become more accustomed to the system. However, subject 2 did not exhibit improved task completion time during the third trial. Instead, they displayed controlled movement without exerting excessive force.

The subsequent two figures offer details regarding the third trial conducted by subjects 1 and 2, aiming to compare their distinct behaviors. These figures encompass several parameters, such as the user-applied force, damping parameters, steering error, distance to the next waypoint (referred to as POI) and the velocities of the right and left wheels. Notably, a significant increase in the distance to the next waypoint indicates the acceptance of a new waypoint, providing additional insight into the subject's progress.

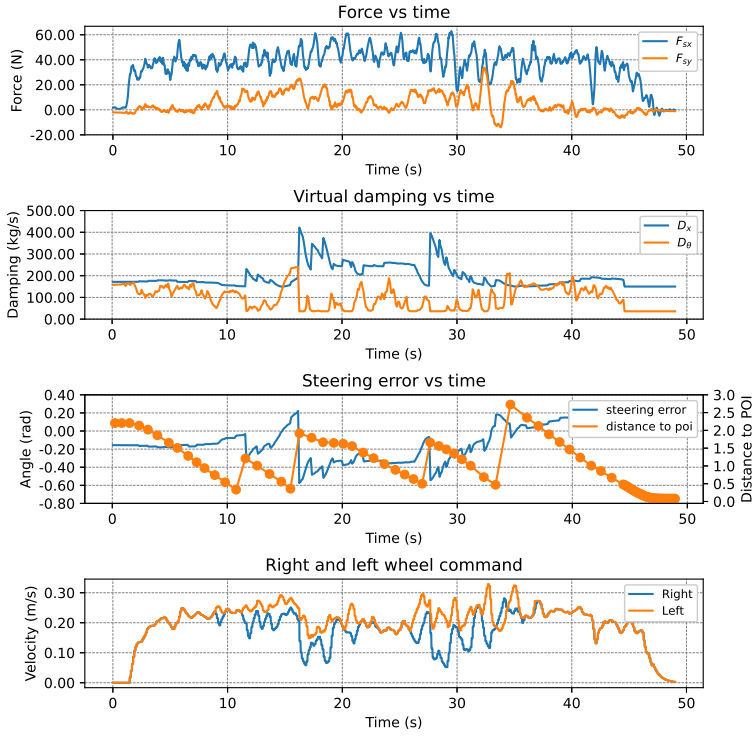


Figure 7.33: Details regarding the third trial conducted by subject 1.

Moreover, the largest difference lies in the magnitude of the applied force. Subject 2 (Figure 7.34) consistently applies a force in the x-direction of approximately 20N with minimal variations, whereas subject 1 (Figure 7.33) applies nearly double the force with greater variability. Consequently, these differences result in distinct behaviors. Subject 1 completes the path quicker but exhibits less accuracy, as indicated by the steering error failing to converge to zero before reaching the next waypoint. In contrast, subject 2 takes longer to navigate the course but demonstrates greater precision, as indicated by the steering error converging to zero before reaching each waypoint.

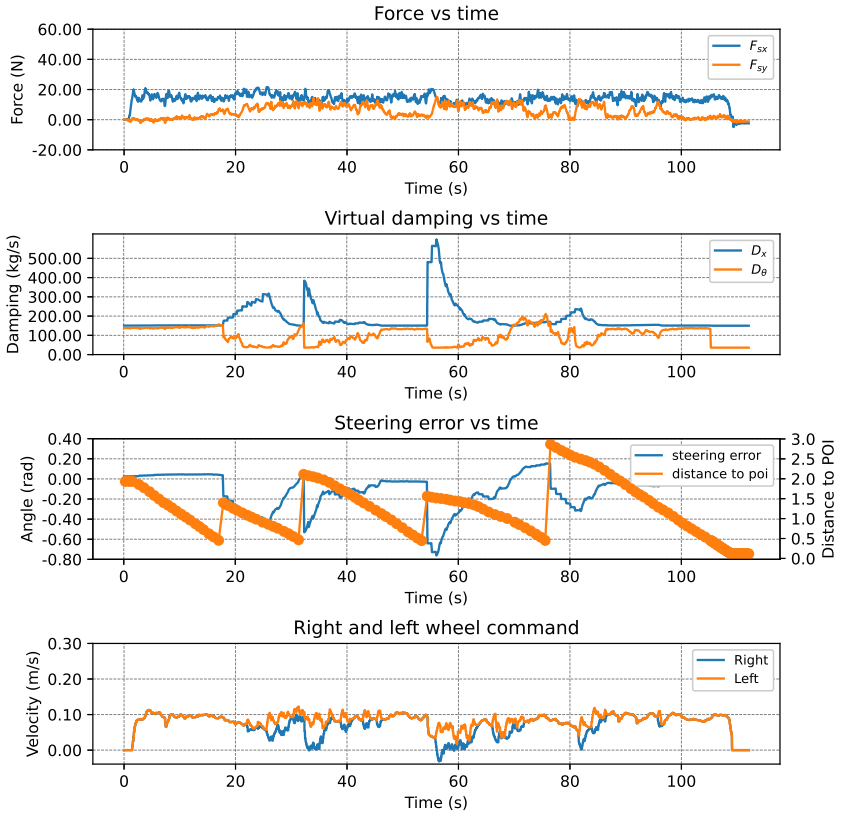


Figure 7.34: Details regarding the third trial conducted by subject 2.

7.5.2 Walking Support

In the walking support experiment, participants are provided with information regarding the location of all waypoints, as outlined in Section 6.4. Given that participants are already familiar with the path and the system after completing the guidance experiment, the previously observed improvement between trials is absent in this experiment. Consequently, only one of the two trials per participant is depicted in Figure 7.30.

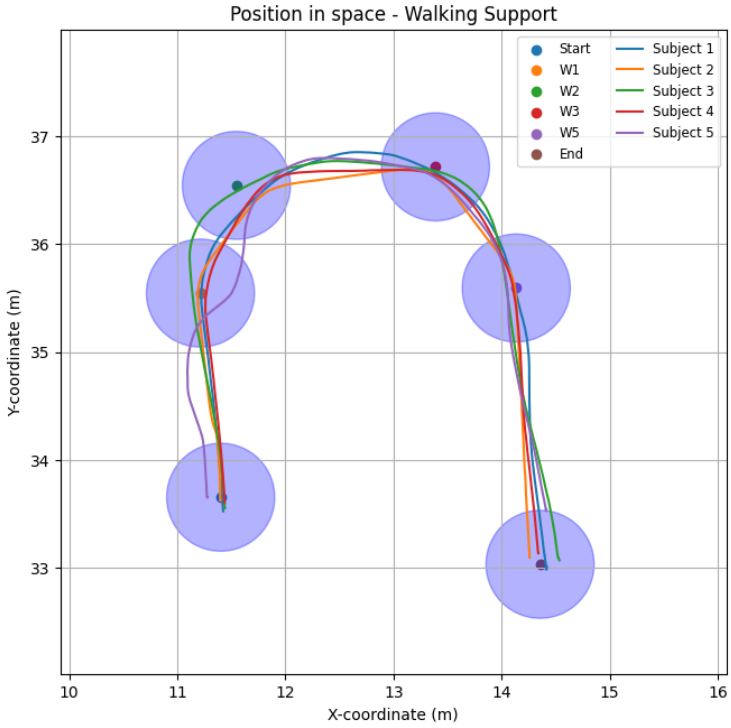


Figure 7.35: The path followed by the subjects during walking support.

Furthermore, the performance metrics are recorded and presented in Table 7.6. The completion time has decreased further compared to the guidance experiment. Additionally, the cross-track error is recorded to facilitate the comparison with the guidance experiment, despite participants not being constrained by the path. The trajectories reveal that participants tend to cut corners compared to the guidance experiment, resulting in greater deviation from the path.

Table 7.6: Performance metrics recorded for the subjects during walking support.

Metric	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5
Correlation	0.32	0.08	0.13	0.58	0.30
Distance [m]	8.9	8.5	9.0	8.5	8.2
Duration [s]	40.7	84.2	40.0	45.3	47.4
Impulse [Ns]	1386.8	1389.7	1407.1	1326.7	1347.5
Work [Nm]	303.9	140.0	316.2	249.1	232.8
XTE [m]	643.1	2395.3	3030.0	4104.9	834.6

Moreover, a comparison between the final trials of guidance and walking support for subject 1 is presented in Figure 7.36. It is evident that the forward velocity in the walking support trial (right) maintains a more consistent pattern compared to the guidance trial (left). Additionally, there is a general decrease in force levels during the walking support trial, particularly noticeable in the force applied in the y-direction, which displays a distinct trend compared to the oscillatory pattern observed in the guidance trial. These distinctions will be elaborated on in the discussion.

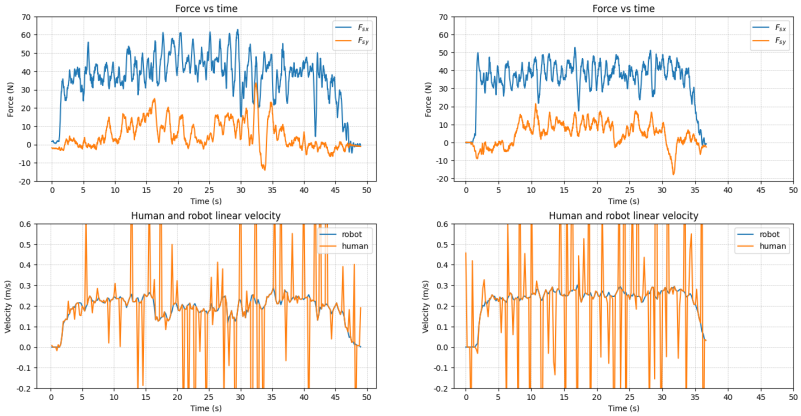


Figure 7.36: Comparison between the guidance (left) and walking support (right) trials for the first subject.

8

Discussion

The discussion delves into the main findings of this study, encompassing the evaluation of both walking support and the guidance solution. It explores the implications of these findings within the context of elderly assistance, identifying existing limitations and proposing potential advancements. Through this structured approach, the aim is to provide a comprehensive understanding of the results and their relevance to force-controlled walking assistance.

8.1 Preliminary Results

When designing the controllers, it is crucial to consider the update rate of the Kompaï robot, the noise and drift of the force sensor, as well as the effects of the admittance parameters. The preliminary results offer insights into these aspects.

8.1.1 Step Response

The step response analysis examines the sensor's update rate and the latency introduced by the Kompaï robot. The sensor operates at 500Hz, which does not impose any limitations on the controller. However, insights derived from the step response (Figure 7.1) suggest that the internal control system of the Kompaï robot operates at approximately 20Hz. Consequently, employing a controller with a frequency exceeding this value does not result in improved transmission.

In terms of latency, the controller with an update rate of 10Hz and a publish rate of 100Hz, as shown in Figure 7.1, demonstrates the lowest time constant. Since the reference is published ten times more frequently than it is updated and five times more frequently than the Kompaï robot's update rate, synchronization is less critical. However, if the publish rate were to match with the Kompaï robot's update rate and synchronization was imperfect, the robot might fetch the published reference just before a new controller update. This could result in a delay of one update period. Therefore, publishing faster than the Kompaï robot's update rate minimizes latency.

Additionally, higher update rates, such as 100Hz in Figure 7.1, cause irregularities in publishing reference velocities. This occurs as the controller's computation time exceeds what is allocated by the update frequency, namely $\Delta t = 1/100\text{s}$. Therefore, the subsequent updates are delayed until the current is computed. Since publishing follows the update, this causes delays between the published velocities, as seen in the figure. Moreover, the impact of a higher update rate on the admittance controller, as discussed later, justifies the selection of a lower update rate together with a higher publish rate.

8.1.2 Force Sensor

Upon analyzing the performance of the force sensor, Figure 7.2 indicates substantial drift along the z-axis. The observed drift influences the precision of force measurements, consequently affecting the robot's decision-making process. A notable implication is the possibility of transitioning from the IDLE to the PRESENT state if the resultant force exceeds the specified threshold. In such a scenario, the drift might induce the transition, even in the absence of force being applied to the handles.

Moreover, the observation is gathered over the course of an entire day to assess the behavior of the force sensor. Notably, the force measurements stabilize over the observed period, likely due to the sensor attaining operational temperature. When utilizing the robot in human interactions where sensor accuracy is crucial, it is recommended to regularly reset the sensor's offset before initiating movement or to allocate adequate time for the force measurements to stabilize.

Furthermore, the force sensor exhibits a degree of noise, comprising both inherent noise and possible bias. The bias, illustrated in Figure 7.3, may arise from drift or offset, particularly if the sensor was not calibrated prior to data collection. However, the amplitude of this noise is minimal, with a variation of 0.15N. Although it represents a disturbance, its influence seems restricted and will not present a substantial challenge to the intended use.

To reconstruct the forces exerted on the handles F^{hl}, F^{hr}, an analytical expression has been formulated (Equation 4.20), which incorporates both known measurements and unknown moment arms. Consequently, the application of known forces to the handles facilitates the computation of moment arms. While determining moment arms in the z-direction is straightforward, accomplishing this in the x-direction poses a more intricate challenge.

As discussed in Langeweg (2023), it is feasible to conduct an experiment that involves disassembling the handle construction and aligning the sensor's x-axis with gravity to apply a known mass. However, this time-consuming process is omitted within the thesis's scope. Instead, approximate measurements were taken and adjusted to generate Figure 7.4, ensuring that a force applied to the left handle minimally influences the force on the right handle. Acknowledging the inherent coupling within the system, it is improbable to achieve zero deviation at the handle opposite to the applied force. Therefore, the results obtained through this simplified approach are deemed adequate for the objectives of the thesis.

8.1.3 Simulation of Admittance Controller

Simulations are conducted to verify the impact of various admittance parameters on the system's behavior. As shown in Figure 7.10, the damping parameter affects the steady-state velocity, while the mass parameter influences the system's inertia. Specifically, a higher mass results in slower acceleration when a force is applied and slower deceleration when the force is removed.

Additionally, the effect of removing the stiffness parameter is illustrated in Figure 7.6, where the velocity converges to zero while the position diverges. This occurs because the transfer function is second-order with one pole at the origin. Consequently, the velocity will converge to zero, but the position will not. This means that the walker will not return to its starting position when the force is no longer applied, which is necessary for providing walking support where the user commands motion.

8.2 Walking Support

In the context of walking support, conducting experiments is essential to identify the most suitable control method and parameters for interactions with elderly users. Comparing the solution from the project report (Langeweg (2023)) with the new controller developed in the current study, supports the decisions made in favor of the new controller.

8.2.1 Turning Strategies

When comparing the various turning strategies based on the controller from the project report (Langeweg (2023)), the constant ω approach proves to be effective. Other solutions maintain a constant forward velocity when initiating turns, which may feel too fast if the turning radius R becomes small. These methods are adequate as long as the turning radius does not decrease significantly, meaning the user does not apply excessive force in the y-direction. In contrast, the constant ω approach is smoother when a large force is applied in the y-direction because the forward velocity decreases as the turning radius increases. However, a limitation of this method is that the forward velocity may abruptly change when turning is initiated.

The challenges outlined above support the decision to develop a more dynamic approach for interaction between the user and the robotic walker. Therefore, the experiments conducted with various turning strategies are compared to the admittance controller developed in this study. As shown by the commanded velocity in Figure 7.7 and the heading angle in Figure 7.9, the admittance controller exhibits a smoother behavior than the turning strategies. Moreover, the initiation of movement and increase in velocity with the admittance controller are dynamic, whereas the turning strategies rely on rule-based increments in velocity.

8.2.2 Admittance Controller

Parameters

An experiment, equivalent to the preliminary simulations of an admittance model, is conducted to analyze the effect of the admittance parameters on the physical system. The results displayed in Figure 7.10 and Figure 7.11 support the same conclusion drawn from the simulations. Higher damping increases the steady-state value given a constant force, while higher mass increases inertia, resulting in a slower system response. While the damping parameter also affects the time constant, prioritizing the selection of mass is crucial. Mass determines the increase in velocity per unit force and should therefore be chosen first to ensure comfortable accelerations for the user. Once the mass is determined, the damping can be selected to achieve the desired time constant or resistance within the stability limits.

As outlined in the methodology, one approach for determining the mass parameter involves ensuring that it achieves the desired change per unit force, as described by Equation 6.31. For future enhancements, it would be advantageous to customize the mass parameter according to each subject's maximal force exertion. In this scenario, the maximum acceptable increase in velocity, given the maximum force, is determined. Subsequently, the mass parameter is computed as a function of this value, the maximum force $F_{\max}$ and the timestep Δt. However, for the sake of simplicity, the mass parameter remains consistent across all subjects in this study.

Update Rate

Similarly to the step response in the preliminary experiments, the experiments concerning the update rate of the admittance controller support the decision to use asynchronous update and publish rates. When both rates are set to 100Hz, as shown in Figure 7.12, irregularities similar to those observed in the step response occur. Since the required computation time exceeds the time provided by the 100Hz update rate, the controller cannot operate efficiently at this high frequency. Consequently, it causes delays in transmitting the reference signal, potentially affecting the interaction dynamics.

To further support the choice of asynchronous update and publish rates of 10Hz and 100Hz, respectively, the interaction dynamics for various rates are evaluated in Figure 7.13. When both rates are set to 100Hz, the interaction becomes oscillatory even when the user attempts to maintain a constant force. These oscillations are likely due to high frequency variations from the force sensor, which operates at 500Hz. With an update rate of 10Hz, the sensor measurements are utilized only ten times per second, effectively filtering out high-frequency variations. Conversely, with an update rate of 100Hz, the sensor measurements update the velocity a hundred times per second, resulting in a more oscillatory velocity. As the velocity fluctuates due to high-frequency variations in force, it induces oscillations in the applied force as the user holds onto the handle. Thus, the high frequency variations are amplified, resulting in oscillatory interaction.

Therefore, the lower update rate effectively filters out the user's intention, as illustrated in Figure 7.15. The force shown in this figure is simulated, serving as an approximation of real interaction. Although it does not precisely replicate real-life interaction forces, it

effectively demonstrates the impact of a lower sampling rate.

Given that the bandwidth requirement for interacting with human operators is not significantly large, especially with the elderly, a 10Hz update rate is considered sufficient. However, for future advancements, it is recommended to introduce a filter for the force measurements independent of the controller. This would ensure that issues with oscillations in velocity are no longer dependent on the controller's sampling rate.

Moreover, the controller's behavior at synchronous rates of 10 Hz appears to perform similarly to the asynchronous rates when a user is interacting with the system. However, as discussed for the step of input force in Figure 7.12, the asynchronous rates exhibit less latency. Consequently, the update and publish rates for the admittance controller are selected to be asynchronous.

Stability Limits

From the expression derived in Equation 6.29 and the experiment shown in Figure 7.16, it is clear that excessively increasing the ratio between damping and mass can destabilize the system. It may seem counterintuitive that increasing the damping parameter too much leads to instability, as an overdamped system generally causes the response to move slowly toward the equilibrium. However, due to the discretization process, the amplification factor for the velocity $v[k-1]$ should not be less than negative 1.

This factor determines how the previous velocity influences the current velocity in the numerical integration process. If this factor is less than negative 1, the velocity will oscillate between negative and positive values and experience amplification, making the system unstable. Therefore, it is crucial to satisfy the stability limit provided in Equation 6.29.

Cost Evaluation

As previously discussed, after selecting the mass parameter the interaction dynamics can be adjusted to achieve the desired training effect or to provide support for weaker individuals. The impact of varying damping parameters, specifically the damping parameter for forward velocity d_x, is examined in Figure 7.17. In this analysis, two different damping parameters are tested alongside the same mass parameter. The interaction dynamics clearly change as a result of the higher damping parameters. The cost evaluation demonstrates that more effort is required to move the robot with higher damping parameters, making it more suitable for training purposes. However, reducing the damping can benefit weaker individuals by making it easier to command motion.

Furthermore, the correlation between the human and the robotic walker suggests that the controller with lower damping parameters performs better. In this case, the whole system moves more cohesively, with the human and the robot working in the same direction. Nevertheless, the experiments involving human subjects have revealed that drawing definitive conclusions from this metric is not feasible. Therefore, relying on it for this experiment may lead to false conclusions.

User Interaction

To facilitate a smooth initiation of movement, a state governs whether the admittance controller is active or inactive. Under this framework, a rule is necessary to transition from the IDLE state to the PRESENT state, activate the controller and release the breaks. Initially, only forces in the z-direction were monitored to detect the user's presence. This behavior arises from users' inclination to apply downward pressure on the handles for support when approaching the robot.

However, as users prepare to initiate motion, they begin to push forwards, potentially decreasing the handle forces in the z-direction. Additionally, users may even apply upward pressure on the handles, generating a positive z-force that exacerbates the issue. Consequently, the robot ceases motion due to the reduction in force, leading to an abrupt halt that reestablishes force in the z-direction, resulting in an oscillatory behavior where the robot repeatedly starts and stops.

To address this concern, the rule was modified to classify the user as present when the resultant force surpasses a predefined threshold, as outlined in Equation 6.10. This adjustment aims to prevent the previous oscillations, as the force in the x-direction increases while the force in the z-direction decreases.

Additionally, conditions were established to account for the forces in the forward (Equation 6.11) and lateral (Equation 6.12) directions, preventing the robot from frequently starting and stopping. This allows the user to remain stationary, with the state being present and the admittance controller activated, without the robot making minor adjustments due to small force variations. Without these conditions, the robot would make slight movements even when the user, present at the handles, does not intend to move.

The study identifies two main strategies for initiating movement. The first involves starting with a sudden jerk to alert the user, while the second initiates movement slowly to allow the user time to adjust.

In this study, the admittance controller employs the slow transition into movement, with the controller gradually building up speed due to the inertia from the mass parameter. For walking support, where the user controls the movement and velocity is driven by the user's intention, a sudden jerk may catch the user off guard and cause a loss of balance. However, during guidance, a sudden jerk in the direction of the next waypoint might help direct the user appropriately. Thus, determining the most effective approach is complex and likely varies based on the application and individual user needs. Consequently, it is advisable to conduct physical experiments with elderly individuals to evaluate the effects of different strategies.

8.3 Fall Prevention

8.3.1 Parameters

Regarding the fall prevention parameters, the same statistical measures are utilized throughout the entire study. However, variations in the mean walking position behind the walker may compromise fall prevention effectiveness. Therefore, for future enhancements, customizing the walking ellipse to each individual user is recommended. Additionally,

as walking statistics are adjusted, the coefficients governing the adjustment of damping parameters change. Consequently, streamlining this process is essential to facilitate a smoother adaptation for new users.

Moreover, as elaborated in Section 7.3.2, the maximum permissible damping parameters are selected to exceed the threshold ensuring the absence of oscillations in velocity, as defined by Equation 6.28. This is essential to provide sufficient resistance for assisting individuals in restoring balance. Thus, these minor oscillations in velocity, particularly when the velocity is low, are not regarded as problematic for the interaction.

Unlike the varying damping parameters, the mass parameters remain constant. An interesting experiment involves adjusting the mass parameters to make the walker feel heavier. While this adjustment makes the walker challenging to move when stationary, changing the mass while the walker is in motion would increase its inertia. As a result, the walker's speed would decrease slowly, offering minimal support to a user losing balance. Therefore, adjustments are limited to the damping parameters alone, to facilitate fall prevention.

8.3.2 Correlation Coefficient

The normal distribution of walker position introduced by Hirata et al. (2006) includes the correlation coefficient, as represented by Equation 4.23. While this approach was designed for a cane robot with omnidirectional drive, the current study employs a walker robot with a differential drive. Consequently, the user is positioned more directly behind the walker compared to a cane-type robot. As a result, when walking in a straight line, the correlation coefficient for the walker robot will be zero.

Additionally, unlike the omnidirectional drive of a cane robot, the differential drive mechanism cannot navigate in all directions. Therefore, the user remains behind the robotic walker when turning. However, the coefficient may exhibit slight variations during turns, as the user may need to step sideways for sharper turns. Consequently, collecting walking statistics during an uneven distribution of left and right turns would render them unrepresentative. Hence, the correlation coefficient is deemed excessive for fall prevention with a robotic walker.

8.3.3 Walking Dynamics

As movement with the robot is primarily forward, it is assumed that all falls will occur forward or sideways. Consequently, the admittance parameters are optimized for these directions. The coefficients governing the adjustment of damping parameters are set to achieve maximum damping when the user is at the maximum distance from the walking ellipse. This maximum distance is calculated for a user positioned behind the mean of the walking ellipse, corresponding to a forward fall.

However, if a backward fall occurs, the user would be positioned in front of the walking ellipse, closer to the walker. In this scenario, the user would exert force on the robotic walker's handles to try to regain balance. As the user's center of mass moves further behind their body, the force on the handles would increase. The damping parameters alone may not suffice to counteract this force, as the user's weight could pull the robot's wheels faster than the motors rotate. To address this, it would be necessary to engage the

robot's wheel brakes, assuming the user has the strength to hold onto the handles.

Moreover, to stop the robot at the correct moment, conditions for imbalance are defined in Equation 6.33 and Equation 6.34. Since the best method for helping users recover balance is not straightforward, most walkers halt when a near fall occurs (Pereira et al. (2019)). Therefore, if the force imbalance is too large or if too much of the user's weight is supported by the handles, the walker stops. This action is intended to support users who are out of balance. It is recommended to conduct physical experiments with elderly individuals to evaluate the effectiveness of this approach.

From the evaluation of the fall prevention strategy, specifically regarding the adjustment of the damping parameter for the forward velocity d_x in Figure 7.20, minor oscillations in the velocity are observed. This occurs when the ratio between damping and mass surpasses the limit recommended by Equation 6.28 to prevent oscillations in velocity. However, increasing damping parameters within this range is considered essential to ensure sufficient support. Although minor oscillations persist, they do not significantly disrupt the user. Therefore, a trade-off is struck between reducing oscillations and providing adequate assistance. Given the urgency of support during emergencies, the presence of oscillations remains acceptable as long as the system maintains its stability.

8.4 Guidance

In this section, the two developed guidance solutions are evaluated based on the conducted experiments. Additionally, the two approaches are compared and their limitations presented.

8.4.1 VAC

Achieving effective haptic sensations that guide users along the correct path depends on the functions adjusting the admittance parameters. Selecting these functions requires a trade-off between user autonomy and deviation from the intended path. Since haptic feedback aims to assist rather than control the user, excessively high damping in proximity to the path should be avoided. However, for elderly individuals with reduced cognitive abilities, prioritizing path proximity over user autonomy may be more appropriate. Therefore, it is acknowledged that different individual capabilities necessitate varying admittance parameters.

Furthermore, the width of the function adjusting the damping parameter for the forward velocity d_x is governed by the coefficient δ_{dx} . Selecting a low value for this coefficient means that even small steering errors make it difficult to move, thereby keeping the user in close proximity to the path. However, when the current waypoint is reached, the steering error increases as the next waypoint is targeted. Consequently, the damping parameter will increase and slow the walker down.

If the user has built up some speed and the δ_{dx} coefficient is low, the transition to the next waypoint will feel abrupt as the robot suddenly slows down due to the high damping. Therefore, it is important to strike a balance between maintaining user proximity to the path and ensuring smooth transitions to new waypoints when selecting the δ_{dx} coefficient.

Additionally, selecting a low minimum damping makes it significantly easier to

navigate toward the correct path. However, the previously mentioned challenge regarding transitions to new waypoints becomes more significant. With a low minimum damping, users will move at a higher velocity as they approach the waypoint. If the δ_{dx} coefficient is not appropriately chosen, the parameters will shift abruptly from a low to a high damping parameter upon reaching the waypoint, resulting in a sudden decrease in speed. Therefore, it is essential to carefully consider the relationship between δ_{dx} and $d_{x,\min}$ when selecting the coefficients.

Moreover, in Figure 7.22, the maximum damping parameter for the forward velocity is chosen to exceed the relation between mass and damping, detailed in Equation 6.28. Consequently, near maximum damping, some oscillations in velocity may occur. This decision aims to increase the resistance for users walking in the wrong direction. As the user deviates further, the walker's speed decreases, with maximum damping resulting in nearly zero velocity. Consequently, minor oscillations are acceptable at such low speeds. However, during transitions between waypoints at higher speeds, oscillations should be avoided. Thus, the selection of the δ_{dx} coefficient should ensure that damping does not reach its maximum when targeting the next waypoint.

In contrast to the damping parameter in the forward direction, the maximum damping parameter for turning d_θ adheres to the inequality specified in Equation 6.28. Consequently, no oscillations are evident in the velocity, as illustrated in Figure 7.26. However, as shown in Figure 7.27, excessive parameters lead to oscillations, making it challenging for the user to identify the correct direction. Therefore, the strategy involves preventing oscillations and adjusting $P_{d\theta}$ to ensure a sharp transition from low damping to high damping. This approach enables the robotic walker to respond swiftly when the user applies force towards the correct direction, while resisting movement in the wrong direction.

8.4.2 Guided Walk

The guided walk refers to the feature provided by the Kompaï robot's path planner. This feature regulates the angular velocity to stay on the path, while the user controls the linear forward velocity. Thus, the user determines the speed at which the path is followed but not the direction. However, if no forward velocity is commanded, the robot will rotate in place to align its heading toward the waypoint. As shown in Figure 7.28, the guided walk occasionally overshoots some waypoints. The path planner aims to maintain a smooth movement causing deviations from the path when waypoints are positioned close together.

Additionally, a waypoint is accepted and a new one is targeted once the walker is within the acceptance radius. With the same set of waypoints, if the radius were lower, the guided walk would not accept the waypoints with an overshoot and would need to backtrack to reach them. Therefore, for paths with closely spaced waypoints, it is important to adjust the acceptance radius accordingly. In open spaces, the acceptance radius should be large enough to prevent the walker from having to return to waypoints. Conversely, in narrow hallways, the walker should move slower to minimize overshoot. Consequently, it is recommended to set speed limits based on the environment or the proximity of the waypoints, to ensure accurate navigation.

8.4.3 Comparison

When comparing the two solutions, it is evident that the guided walk offers less flexibility than the VAC. The objective of the VAC is to provide the user with a sense of control while making it more difficult to deviate from the path, thereby emphasizing the preferred route. In contrast, the user cannot change the trajectory during the guided walk and can only control the speed at which they traverse the path.

Moreover, since the user is responsible for maneuvering during VAC, it requires cognitive capabilities, making it a better tool for maintaining autonomy. Additionally, when navigating narrow hallways, the user can adjust the path if necessary, allowing them to move around obstructions. In contrast, with the guided walk solution, the user can only stop the walker and cannot correct its motion to avoid collisions. Furthermore, with appropriate parameters, the VAC generally tracks straight line segments better than the guided walk, which may overshoot. Thus, in confined spaces, the VAC is the preferred solution.

On the contrary, if the objective is to navigate between locations in a less demanding environment, opting for the guided walk may be more appropriate. This choice reduces the user's cognitive load, as the robot manages the trajectory. Furthermore, the required level of work is notably lower compared to the VAC. Consequently, both cognitive and physical demands are significantly reduced in comparison to the VAC. Therefore, it could provide mobility to individuals with physical or cognitive impairments, who are unable to move independently. Additionally, the user's velocity is smoother and better synchronized with the walker's speed, potentially enhancing the overall user experience.

8.5 Experiments with Human Subjects

8.5.1 Guidance

Effect of Training

When comparing the first and third trials in Figure 7.30, it is evident that the performance of nearly all participants has improved. However, the performance of subject 4 was already adequate in the first trial, resulting in minimal to no improvement in the third trial. This is likely due to the subject's familiarity with the system, as they regularly work with the Kompaï and are well-acquainted with its capabilities.

Moreover, the first trial for the remaining subjects exhibits poor performance, as they all appear to go straight past the first waypoint without turning. This suggests that the variable admittance parameters are inadequate when the user is unfamiliar with the system. Additionally, the subjects were instructed to navigate from the starting position to the end position without knowing the waypoints, as outlined in Figure 6.6. However, their behavior indicates an anticipation for the robot to execute the turns on their behalf. Consequently, it reduces their initiative to actively seek the path of least resistance.

As the users decide the trajectory, they may ignore the cues from increased resistance and deviate from the intended path, particularly if the user expects the robot to perform the turns. Therefore, the control solution designed to encourage user autonomy while emphasizing the intended path, may fail if the user is not actively seeking the easiest route.

However, by the third and fourth waypoints in the first trial, it is evident that there is an improvement in performance. Following this trend, there is a notable improvement in the performance of all subjects in the third trial, as evidenced by their ability to command smooth trajectories. This implies that once the user becomes acquainted with the system, it behaves as intended. However, determining whether this enhancement is attributed to the user's familiarity with the system or their knowledge of the intended path is challenging.

This raises further concerns about employing the technology for elderly individuals with cognitive impairments such as dementia. If familiarity with the system is required to achieve adequate performance, these patients will need to become accustomed to it every time it is employed. Hence, it is recommended to conduct various experiments using different routes to assess the influence of system familiarity on performance. Additionally, evaluating the rate at which the learning effect diminishes, especially among elderly individuals, would provide valuable insights for further enhancements.

Evaluation of Metrics

The experiments reveal two distinct user behaviors, as detailed in Section 7.5.1. The main distinction arises from the application of varying levels of force, resulting in significant differences in completion time. These disparities pose challenges in comparing subjects. Additionally, the subject applying minimal force carefully aligns the robot's heading to eliminate steering errors between each waypoint. Consequently, the behavior observed in Figure 7.34 appears more precise than that of the subject applying greater force in Figure 7.33. However, due to the longer duration taken by the subject applying less force, the accumulation of cross-track error over time leads to a larger overall error. Consequently, some metrics present difficulties in accurately evaluating performance.

Similarly, the correlation between human and robot speed did not perform as expected. In the comparison between the guided walk and the VAC, the correlation coefficient appeared to be a useful indicator. However, in experiments involving human subjects, no definitive conclusions can be drawn from the low correlation coefficient. This may be due to these subjects walking with less confidence, causing them to lag slightly behind the walker to better understand its movement. Therefore, it is recommended to conduct additional tests with this metric to evaluate its reliability and effectiveness.

Moreover, the impulse metric significantly decreased from the first to the third trial, suggesting that less overall change in momentum occurred. However, since time was considerably reduced and impulse is a function of force and time, the decrease in impulse might be attributed to the reduced time. Nonetheless, the impulse for subject 2 decreased compared to the first trial, even though the subject spent the same amount of time on the last trial. This may indicate that, as users become more familiar with the system, they apply less excessive force. Additionally, the work performed shows that less force was applied to navigate the same path in the last trial. However, for one subject, the work increased considerably, likely because they became familiar with the path and increased their speed to complete it as quickly as possible. Therefore, for future experiments, it is recommended not to encourage subjects to prioritize speed.

Waypoints

An advantage of using the LoS controller is that it permits the use of sparse waypoints. As such, the waypoints employed during the experiments are spaced relatively sparsely. However, since straight-line segments connect the waypoints, a limited number of waypoints results in a non-smooth path. While adding more waypoints facilitates a smoother path, incorporating them may be cumbersome. Hence, the selection of waypoints necessitates a trade-off between simplicity and smoothness.

8.5.2 Walking Support

When comparing the walking support experiment with the guidance, it is clear that the completion time is further reduced. Additionally, the impulse is significantly lower than in the guidance trials. This is likely due to the subjects being aware of the waypoint locations and aim to command smoother motions. Furthermore, the damping parameters are not abruptly increased upon reaching the waypoints, unlike in the guidance trials. As discussed earlier, sudden increases in damping reduce the speed abruptly and induce force in the handles. Therefore, without the guidance activated, the turns are smoother. This is also reflected in the reduced work exerted by each subject compared to the guidance trials.

8.6 Limitations

This study's limitations are primarily related to the significant differences in physical and cognitive abilities among the elderly population. When designing robotic systems, it is crucial to consider this variability to ensure effective and safe assistance. One challenge is selecting the appropriate admittance parameters. Balancing the need for user autonomy with the necessity of staying on the correct path depends on the individual user's requirements. Addressing these challenges is essential for creating a system that improves user experience.

One aspect to consider is the effect of cognitive impairments. As previously mentioned, individuals with cognitive impairments, such as dementia, may find it difficult to use the guidance solution. Familiarity with the system appears to be necessary for adequate performance. Therefore, these individuals might never learn the system well enough to use it effectively.

Furthermore, muscle impairments can complicate the detection of balance loss. While healthy individuals typically distribute their weight evenly on each handle, those with conditions such as hip pain may rely more on the unimpaired side. As a result, a single solution for all users may be less effective in detecting falls. Hence, future research should prioritize the development of multiple modes and the enhancement of adaptability, to offer customized solutions that address individual needs.

The limited variety of subjects and experiments is acknowledged as a constraint in evaluating the system's performance. For instance, delving into the effects of higher damping on individuals with Parkinson's disease could prove insightful. The patients with Parkinson's experience involuntary movements. Therefore, higher damping might aid them in controlling the robotic walker, as it could mitigate the impact of tremors and

uncontrollable motions.

Additionally, the study acknowledges the complexity of experimental design. For example, isolating the influence of variable admittance parameters on guidance is proven to be challenging. To accurately assess performance, subjects should be unaware of the path beforehand. However, the outcome might be influenced by the effect of training and familiarity with the path when conducting multiple trials. Furthermore, the proposed metrics are recognized as incomplete for making definitive conclusions. Consequently, it is advisable to allocate more time to experimental design, to thoroughly evaluate the effectiveness of the implemented solution.

8.7 Future Work

In addition to the enhancements discussed earlier, this section delves into further suggestions to improve the interaction between the user and the robot.

To address the limitations associated with the guidance solution for cognitively impaired individuals, future research should aim to make interaction more intuitive. This could involve integrating an additional layer to the controller in order to manage user interaction. For example, this layer could introduce a subtle jerk in the intended direction to signal the initiation of a turn during guidance. Furthermore, visual feedback, such as displaying directional arrows, could be implemented to further assist users.

Additionally, the effectiveness of the proposed criteria for detecting imbalance is likely dependent on individual needs. For example, individuals with Parkinson's disease may exhibit uncontrolled tremors that impact the forces applied to the handles. Similarly, patients with hip impairments might distribute their weight unevenly on one side of the walker. Therefore, a more adaptable approach is recommended. A potential strategy, as presented by Li, Akiyama, et al. (2021), involves utilizing gait phase estimation to identify abnormal gait patterns. Upon detecting an abnormal gait, fall prevention measures could be initiated.

Another aspect to consider is the walker's ability to facilitate training. As such, adjusting the virtual mass could increase resistance, necessitating greater muscle effort to generate propulsion force (Li, Y. Yamada, K. Yamada, et al. (2022)). Consequently, this modification increases the training impact on the targeted muscle group. Additionally, introducing different modes, such as a sport mode, may facilitate training for individuals with higher fitness levels. However, given the potential increase in speed, it may be advantageous to implement a steering lock mechanism at higher velocities, to mitigate the risk of balance loss.

Furthermore, the existing solution for walking assistance may necessitate a lateral stepping strategy during sharp turns. It is advisable to investigate whether this correlates with an elevated risk of falls. Such investigation could guide the formulation of an optimal turning strategy. For instance, it is worth exploring the possibility of regulating the turning radius while employing admittance control.

In addition to utilizing haptic feedback for guidance, the same sensation could be employed for obstacle avoidance. Considering the difficulties that cognitively impaired individuals may encounter in navigating the walker without collisions, obstacle avoidance would enhance safety measures. Accordingly, the vicinity of obstacles could be rendered

as an area with increased resistance. Consequently, the increased resistance would decelerate the user's velocity and direct them away from potential collision trajectories.

The advancement of robotics in elderly care necessitates a thorough comprehension of the users' requirements. Performing experiments is essential to refine the system and ensure user trust and safety. Thus, future improvements should focus on experimental design and the development of valuable metrics to evaluate performance. This approach is essential for the system to achieve a balance between providing meaningful support and preserving the privacy and independence of elderly individuals.

9

Conclusion

The aim of the present research was to investigate and evaluate various force control strategies designed specifically for elderly mobility. By providing guidance and walking support, the primary objective was to facilitate compliance between the user and the robot, while ensuring safety through fall prevention measures. The introduction highlighted the need for intelligent and adaptable mobility solutions, especially for those with weakened muscles. Consequently, integrating service robots in healthcare, such as Kompaï, emerges as a promising method to improve the overall quality of care.

The present study utilized an admittance controller to achieve compliance in human-robot interaction. As such, a state machine governing the logic of user intentions was applied alongside the admittance controller to facilitate walking support. Furthermore, variable admittance parameters were employed to enable haptic sensations that provided guidance to the users. Additionally, to ensure user safety, fall prevention measures were implemented to help the user regain balance. This was accomplished by increasing the admittance parameters when detecting an emergency state, thereby providing additional support.

Finally, experiments with human subjects were conducted to evaluate the effectiveness of the proposed solution. In these experiments, the guidance solution was employed without the user having prior knowledge of the path. Consequently, the impact of the variable admittance parameters was isolated and the performance evaluated.

This study identifies the impact of various admittance parameters on user interaction and stability. Additionally, the utilization of haptic feedback is promising in facilitating guidance while preserving the user's sense of control. The solution enables the user to command motion while haptic feedback emphasizes the intended route. Furthermore, increased damping as a fall prevention measure provides additional support during emergencies.

The effectiveness of most mobility aids is limited due to the diverse range of physical and cognitive abilities among the elderly. Therefore, the study emphasizes the necessity for careful consideration and adaptation to accommodate for various impairments and diseases.

In conclusion, the research demonstrates the promising role of the Kompaï robot as an intelligent mobility aid for elderly individuals. Beyond providing physical support, it fosters user engagement and independence. Future endeavors should address the identified limitations and refine the proposed solution. Additionally, experiments involving representative subjects are imperative to ensure user trust and effectiveness in real-world scenarios.

Bibliography

Langeweg, Martijn (Dec. 18, 2023). "Service robot with force controlled walking assistance". Project thesis. Norwegian University of Science and Technology.

Meld. St. 7 (2019–2020) - regjeringen.no (2023). URL: https://www.regjeringen.no/no/dokumenter/meld.-st.-7-20192020/id2678667/?ch=9 (visited on 12/16/2023).

Naeem, Mohamed A and Samy FM Assal (Jan. 1, 2022). "Development of a 4-DOF cane robot to enhance walking activity of elderly". In: *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 236.2. Publisher: IMECHE, pp. 1169–1187.

Xing, Hongjun, Ali Torabi, Liang Ding, Haibo Gao, Zongquan Deng, Vivian K. Mushahwar, and Mahdi Tavakoli (Apr. 2021). "An admittance-controlled wheeled mobile manipulator for mobility assistance: Human–robot interaction estimation and redundancy resolution for enhanced force exertion ability". In: *Mechatronics* 74, p. 102497.

ISO 8373:2021, Robotics (2023). URL: https://www.iso.org/obp/ui/#iso:std:iso:8373:ed-3:v1:en (visited on 09/28/2023).

Bilgin, Nurdan, Tolga Tutkan, Yılmaz Can Er, and Emre Nayır (2023). "Lateral Support Mechanisms for Smart Walkers to Prevent Sideways Rollover". In: *New Advances in Mechanisms, Transmissions and Applications*. Ed. by Med Amine Laribi, Carl A. Nelson, Marco Ceccarelli, and Saïd Zeghloul. Mechanisms and Machine Science. Cham: Springer Nature Switzerland, pp. 229–239.

Ilic, Irena, Branko Ristic, Ivan Stojadinovic, and Milena Ilic (Aug. 24, 2023). "Epidemiology of Hip Fractures Due to Falls". In: *Medicina* 59.9, p. 1528.

Cerqueira, Rodolfo, José Pereira, Nuno Ferrete Ribeiro, and Cristina P. Santos (Apr. 2023). "Engineering a Safer Future: Electronic Design and Validation of a Cane-Type Robot for Fall Prevention". In: *2023 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*. 2023 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC). ISSN: 2573-9387, pp. 193–198.

Pereira, Ana, Nuno Ferrete Ribeiro, and Cristina P. Santos (Feb. 2019). "A Survey of Fall Prevention Systems Implemented on Smart Walkers". In: *2019 IEEE 6th Portuguese*

Meeting on Bioengineering (ENBENG). 2019 IEEE 6th Portuguese Meeting on Bioengineering (ENBENG), pp. 1–4.

Mahdi, Anas, Jonathan Feng-Shun Lin, and Katja Mombaur (Aug. 2022). "Maintaining mobility in older age - design and initial evaluation of the robot SkyWalker for walking & sit-to-stand assistance". In: *2022 9th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob)*. 2022 9th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob). ISSN: 2155-1782, pp. 01–08.

Zhao, Xiaoyang, Zhi Zhu, Mingshan Liu, Chongyu Zhao, Yafei Zhao, Jia Pan, Zheng Wang, and Chuan Wu (2020). "A Smart Robotic Walker With Intelligent Close-Proximity Interaction Capabilities for Elderly Mobility Safety". In: *Frontiers in Neurorobotics* 14.

Itadera, Shunki, Jun Nakanishi, Yasuhisa Hasegawa, Toshio Fukuda, Masanori Tanimoto, and Izumi Kondo (Jan. 1, 2020). "Admittance control based robotic clinical gait training with physiological cost evaluation". In: *Robotics and Autonomous Systems* 123, p. 103326.

KOMPAI-Assist | Kompai Robotics (2023). KOMPAÏ robotics. URL: https://www.kompai.com/kompai-assist (visited on 12/16/2023).

Jiménez, Mario F., Matias Monllor, Anselmo Frizera, Teodiano Bastos, Flavio Roberti, and Ricardo Carelli (June 1, 2019). "Admittance Controller with Spatial Modulation for Assisted Locomotion using a Smart Walker". In: *Journal of Intelligent & Robotic Systems* 94.3, pp. 621–637.

PPM Robotics AS (2023). URL: https://www.ppm.no/ (visited on 12/16/2023).

Falls and Fractures in Older Adults (Sept. 12, 2022). *Falls and Fractures in Older Adults: Causes and Prevention*. National Institute on Aging. URL: https://www.nia.nih.gov/health/falls-and-falls-prevention/falls-and-fractures-older-adults-causes-and-prevention (visited on 05/29/2024).

Vaishya, Raju and Abhishek Vaish (Jan. 24, 2020). "Falls in Older Adults are Serious". In: *Indian Journal of Orthopaedics* 54.1, pp. 69–74.

Hirata, Y., A. Muraki, and K. Kosuge (May 2006). "Motion control of intelligent passive-type Walker for fall-prevention function based on estimation of user state". In: *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006. ISSN: 1050-4729, pp. 3498–3503.

Fossen, Thor, Morten Breivik, and Roger Skjetne (Sept. 1, 2003). "Line-of-Sight Path Following of Underactuated Marine Craft". In.

FlexGui 4.0 (Dec. 14, 2022). original-date: 2021-04-20T08:56:02Z.

Chang, Yeong-Hwa, Nilima Sahoo, Jing-Yuan Chen, Shang-Yi Chuang, and Hung-Wei Lin (Jan. 2021). "ROS-Based Smart Walker with Fuzzy Posture Judgement and Power Assistance". In: *Sensors* 21.7. Number: 7 Publisher: Multidisciplinary Digital Publishing Institute, p. 2371.

Ding, Damin, Yagang Wang, Houji Zhang, Qiaoling Meng, Hongliu Yu, Yanzhao Lin, and Qiu Chen (Aug. 2023). "ReRobo Walker: Robotic Walker with Fall Detection and Active Safety". In: *2023 IEEE 18th Conference on Industrial Electronics and*

Applications (ICIEA). 2023 IEEE 18th Conference on Industrial Electronics and Applications (ICIEA). ISSN: 2158-2297, pp. 1562–1567.

Ding, Liang, Hongjun Xing, Ali Torabi, Javad K. Mehr, Mojtaba Sharifi, Haibo Gao, Vivian K. Mushahwar, and Mahdi Tavakoli (Aug. 2022). "Intelligent assistance for older adults via an admittance-controlled wheeled mobile manipulator with task-dependent end-effectors". In: *Mechatronics* 85, p. 102821.

Itadera, Shunki and Gordon Cheng (May 2022). "Admittance Model Optimization for Gait Balance Assistance of a Robotic Walker: Passive Model-based Mechanical Assessment". In: *2022 International Conference on Robotics and Automation (ICRA)*. 2022 International Conference on Robotics and Automation (ICRA), pp. 7014–7020.

Itadera, Shunki, Emmanuel Dean-Leon, Jun Nakanishi, Yasuhisa Hasegawa, and Gordon Cheng (Oct. 2019). "Predictive Optimization of Assistive Force in Admittance Control-Based Physical Interaction for Robotic Gait Assistance". In: *IEEE Robotics and Automation Letters* 4.4. Conference Name: IEEE Robotics and Automation Letters, pp. 3609–3616.

Itadera, Shunki, Taisuke Kobayashi, Jun Nakanishi, Tadayoshi Aoyama, and Yasuhisa Hasegawa (Jan. 2019). "Impedance Control based Assistive Mobility Aid through Online Classification of User's State". In: *2019 IEEE/SICE International Symposium on System Integration (SII)*. 2019 IEEE/SICE International Symposium on System Integration (SII). ISSN: 2474-2325, pp. 243–248.

Yokota, M., S. Kawazoe, D. Chugo, S. Muramatsu, S. Yokota, H. Hashimoto, T. Katayama, Y. Mizuta, and A. Koujina (Jan. 2019). "Standing Assistance Control considering with Posture Tolerance of its User". In: *2019 IEEE/SICE International Symposium on System Integration (SII)*. 2019 IEEE/SICE International Symposium on System Integration (SII). Paris, France: IEEE, pp. 543–547.

B, Praveena, C. Vimala Josphine, Nithya Jenev J, Anitha G, Chairma Lakshmi K R, and Vaishnavi G (May 2023). "Robotic Accident Prevention and Alert System for Visually Impaired". In: *2023 International Conference on Disruptive Technologies (ICDT)*. 2023 International Conference on Disruptive Technologies (ICDT), pp. 672–675.

Wakita, Kohei, Jian Huang, Pei Di, Kosuke Sekiyama, and Toshio Fukuda (Feb. 2013). "Human-Walking-Intention-Based Motion Control of an Omnidirectional-Type Cane Robot". In: *IEEE/ASME Transactions on Mechatronics* 18.1. Conference Name: IEEE/ASME Transactions on Mechatronics, pp. 285–296.

Batlle, Joaquim and Ana Barjau (Apr. 1, 2009). "Holonomy in mobile robots". In: *Robotics and Autonomous Systems* 57, pp. 433–440.

Falls - WHO (2024). URL: https://www.who.int/news-room/fact-sheets/detail/falls (visited on 03/12/2024).

Abdalla, Pedro Pugliesi, Lucimere Bohn, Leonardo Santos Lopes da Silva, André Pereira dos Santos, Marcio Fernando Tasinafo Junior, Ana Claudia Rossini Venturini, Anderson dos Santos Carvalho, David Martinez Gomez, Jorge Mota, and Dalmo Roberto Lopes Machado (Dec. 18, 2021). "Identification of muscle weakness in older adults from normalized upper and lower limbs strength: a cross-sectional study". In: *BMC Sports Science, Medicine and Rehabilitation* 13, p. 161.

Li, Pengcheng, Yoji Yamada, Kazunori Yamada, and Mayu Yokoya (2022). “Functional Resistance Training With Gait Phase-Dependent Control Using a Robotic Walker: A Pilot Study”. In: *IEEE Access* 10. Conference Name: IEEE Access, pp. 64976–64988.

Li, Pengcheng, Yoji Yamada, Xianglong Wan, Yasushi Uchiyama, Wakako Sato, Kazunori Yamada, and Mayu Yokoya (June 2019). “Gait-phase-dependent control using a smart walker for physical training”. In: *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*. 2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR). ISSN: 1945-7901, pp. 843–848.

Woollacott, Marjorie, Barbara Inglin, and Diane Manchester (1988). “Response Preparation and Posture Control Neuromuscular Changes in the Older Adulta”. In: *Annals of the New York Academy of Sciences* 515.1, pp. 42–53.

McRuer, D.T. and H.R. Jex (Sept. 1967). “A Review of Quasi-Linear Pilot Models”. In: *IEEE Transactions on Human Factors in Electronics* HFE-8.3. Conference Name: IEEE Transactions on Human Factors in Electronics, pp. 231–249.

McRuer, D. (May 1, 1980). “Human dynamics in man-machine systems”. In: *Automatica* 16.3, pp. 237–253.

Bachelder, Edward and Bimal Aponso (Sept. 22, 2019). “Novel Methods for Estimating Bandwidth and Stability Margins of Pilot-in-Loop Systems”. In.

Bates, J. a. V. (May 1, 1947). “Some characteristics of a human operator”. In: *Journal of the Institution of Electrical Engineers - Part IIA: Automatic Regulators and Servo Mechanisms* 94.2. Publisher: IET Digital Library, pp. 298–304.

Akalin, Neziha, Annica Kristoffersson, and Amy Loutfi (Feb. 2022). “Do you feel safe with your robot? Factors influencing perceived safety in human-robot interaction based on subjective and objective measures”. In: *International Journal of Human-Computer Studies* 158, p. 102744.

Servaty, Ricarda, Annalena Kersten, Kirsten Brukamp, Ralph Möhler, and Martin Mueller (Sept. 2020). “Implementation of robotic devices in nursing care. Barriers and facilitators: an integrative review”. In: *BMJ Open* 10.9, e038650.

Feil-Seifer, David and Maja J. Matarić (2009). “Human RobotHuman–robot interaction (HRI)InteractionInteractionhuman robot”. In: *Encyclopedia of Complexity and Systems Science*. Ed. by Robert A. Meyers. New York, NY: Springer, pp. 4643–4659.

Khan, Said G., Guido Herrmann, Mubarak Al Grafi, Tony Pipe, and Chris Melhuish (Sept. 2014). “Compliance Control and Human–Robot Interaction: Part 1 — Survey”. In: *International Journal of Humanoid Robotics* 11.3, p. 1430001.

Hogan, N. (Aug. 1984). “Adaptive control of mechanical impedance by coactivation of antagonist muscles”. In: *IEEE Transactions on Automatic Control* 29.8. Conference Name: IEEE Transactions on Automatic Control, pp. 681–690.

Peternel, Luka, Nikos Tsagarakis, and Arash Ajoudani (July 2017). “A Human–Robot Co-Manipulation Approach Based on Human Sensorimotor Information”. In: *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 25.7. Conference Name: IEEE Transactions on Neural Systems and Rehabilitation Engineering, pp. 811–822.

Dudek, Gregory and Michael Jenkin (Jan. 1, 2010). *Computational Principles of Mobile Robotics*.

Al-Aama, Tareef (July 2011). “Falls in the elderly”. In: *Canadian Family Physician* 57.7, pp. 771–776.

Bishop, Christopher M. (2006). *Pattern recognition and machine learning*. Information science and statistics. New York: Springer.

Rousseeuw, Peter J. and Bert C. van Zomeren (1990). "Unmasking Multivariate Outliers and Leverage Points". In: *Journal of the American Statistical Association* 85.411. Publisher: [American Statistical Association, Taylor & Francis, Ltd.], pp. 633–639.

Finch, W. Holmes (July 5, 2012). "Distribution of Variables by Method of Outlier Detection". In: *Frontiers in Psychology* 3, p. 211.

Applied Multivariate Statistical Analysis (2024). PennState - Eberly Collage of Science. URL: https://online.stat.psu.edu/stat505/lesson/4/4.3 (visited on 04/04/2024).

1.3.6 Probability Distributions (2024). Engineering Statistics Handbook - National Institute of Standards and Technology. URL: https://www.itl.nist.gov/div898/handbook/eda/section3/eda362.htm (visited on 04/04/2024).

Morris, Dan, Hong Tan, Federico Barbagli, Timothy Chang, and Kenneth Salisbury (Mar. 2007). "Haptic Feedback Enhances Force Skill Learning". In: *Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07)*. Second Joint EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (WHC'07), pp. 21–26.

Abbink, D. A. and M. Mulder (Mar. 3, 2009). "Exploring the Dimensions of Haptic Feedback Support in Manual Control". In: *Journal of Computing and Information Science in Engineering* 9.11006.

Munoz Ceballos, Nelson David, Jaime Alejandro, and Nelson Londono (Mar. 1, 2010). "Quantitative Performance Metrics for Mobile Robots Navigation". In: *Mobile Robots Navigation*. Ed. by Alejandra Barrera. InTech.

Steinfeld, Aaron, Terrence Fong, David Kaber, Michael Lewis, Jean Scholtz, Alan Schultz, and Michael Goodrich (Mar. 2, 2006). "Common metrics for human-robot interaction". In: *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*. HRI06: International Conference on Human Robot Interaction. Salt Lake City Utah USA: ACM, pp. 33–40.

Mondoloni, Stephane, Sip Swierstra, and Mike Paglione (Jan. 1, 2005). "Assessing Trajectory Prediction Performance – Metrics Definition". In: pp. 3.C.1–31.

ROS: Home (2023). URL: https://www.ros.org/ (visited on 12/01/2023).

Nakagawa, Shotaro, Shunki Itadera, Yasuhisa Hasegawa, Kosuke Sekiyama, Toshio Fukuda, Pei Di, Jian Huang, and Qiang Huang (Aug. 2015). "Virtual friction model for control of cane robot". In: *2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. 2015 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), pp. 128–133.

Weik, Martin H. (2001). "time constant". In: *Computer Science and Communications Dictionary*. Ed. by Martin H. Weik. Boston, MA: Springer US, pp. 1786–1786.

Li, Pengcheng, Yasuhiro Akiyama, Xianglong Wan, Kazunori Yamada, Mayu Yokoya, and Yoji Yamada (Jan. 2021). "Gait Phase Estimation Based on User–Walker Interaction Force". In: *Applied Sciences* 11.17. Number: 17 Publisher: Multidisciplinary Digital Publishing Institute, p. 7888.

Appendix

A Scopus

```
elderly
AND
( ( assistive AND walking ) OR ( ( smart OR robotic ) AND walker ) OR
    mobility )
AND
( ( force OR admittance OR impedance ) AND control ) OR ( ( fall AND
    prevention ) )
AND
robot
AND NOT
exoskeleton OR exosuit OR wheelchair OR prosthesis OR humanoid OR
    wearable
```

B User Manual

User Manual for the Kompaï Robot

Assumptions

This user manual assumes a working setup of ROS on a computer running Ubuntu. If you do not have this setup, please follow the instructions at the following URLs:

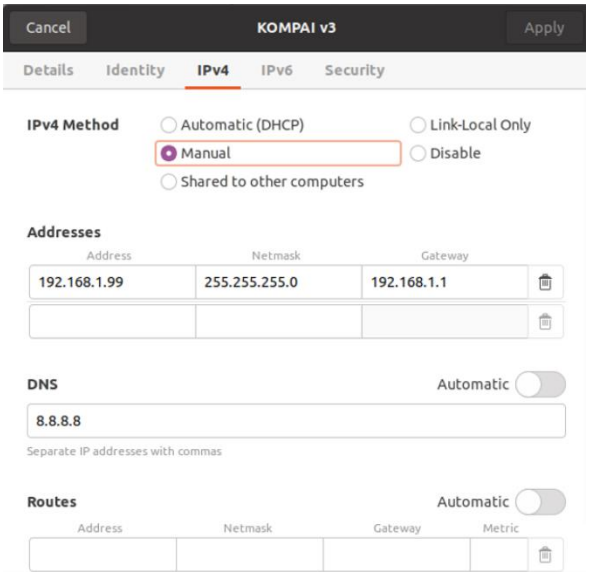
- noetic/Installation: https://wiki.ros.org/noetic/Installation
- Install Ubuntu Desktop: https://ubuntu.com/tutorials/install-ubuntu-desktop#1-overview

Kompaï's interface

Connect to Kompaï's ROS interface

To connect to Kompaï's ROS interface, follow these steps to configure the network after connecting to the "KOMPAI v3" Wi-Fi:

1. Open the Wi-Fi settings and configure the IPv4 settings.
2. Set the **IPv4 Method** to *Manual.*
3. Configure Addresses as follows:
   - **Address:** Choose an IP address for your computer in the format 192.168.1.x, where x must be different from Kompaï's IP address ($x = 2$).
   - **Netmask:** 255.255.255.0
   - **Gateway:** 192.168.1.1
4. Disable automatic DNS.
   - **DNS:** 8.8.8.8
5. On Linux, you may need to toggle the Wi-Fi off and on for the new settings to take effect.



Establish Kompaï as the ROS Master

Modify your `.bashrc` file by opening it with the command **`gedit .bashrc`** in the terminal from your home directory. Then, append the following lines to the `.bashrc` file.

```
export ROS_MASTER_URI=http://192.168.1.2:11311
export ROS_IP=your_ip
```

Replace `your_ip` with the IP address you selected during the Wi-Fi configuration.

The Force Sensor

Connect over Ethernet

To establish a connection with the sensor, follow the instructions outlined in the Quick Start guide for the force/torque sensor:

- Quick Start Guide: https://www.ati-ia.com/app_content/documents/9610-05-1022%20Quick%20Start.pdf

In the default setting, the sensor's IP address is configured as 192.168.1.1. In the Quick Start manual, you are instructed to configure your Ethernet settings as follows:

- **Netmask:** 255.255.255.0
- **IP address:** 192.168.1.100

However, since this falls within the same subnet as the Kompaï's IP address at 192.168.1.1, you will not be able to connect both the sensor via Ethernet and the robot via WiFi simultaneously. Consequently, it is essential to adjust the IP configuration of the sensor and your Ethernet settings to a different subnet, requiring a change in one of the first three octets. For instance, you could select:

- **Sensor's IP address:** 192.168.2.90
- **Ethernet address:** 192.168.2.100

Sensor driver – netft_utils

To obtain a ROS node for the ATI force/torque sensor, install the `netft_utils` package. This package can be installed from the following location:

- netft_utils: https://wiki.ros.org/netft_utils

To configure it to the force sensor used on the robot, modify line 151/152 in the `netft_rdt_driver.cpp` file to:

```
static const double counts_per_force = 1000;
static const double counts_per_torque = 1000;
```

Following this, source the `setup.bash` file with **`source devel/setup.bash`** from the workspace folder, then launch the node using **`rosrun netft_utils netft_node ip_address`**, where `ip_address` corresponds to the sensor's IP address.

Launch the implemented solutions

Repository structure

```
.
├── controller_srvs            # Custom service type to adjust parameters
│   ├── srv                    # Service definitions
│   │   └── EditParameters.srv
├── force_controller           # Package with controller solutions
│   ├── config                 # Configuration files
│   ├── launch                 # Launch files
│   ├── src                    # Source code
│   │   ├── controller.admittance.py
│   │   ├── controller.fall_prevention.py
│   │   ├── controller.guided_walk.py
│   │   ├── controller.turning_strategies.py
│   │   └── utils.py
├── force_transformation       # Package managing the force sensor
│   ├── config                 # Configuration files
│   ├── launch                 # Launch files
│   ├── src                    # Source code
│   │   ├── rotation.py              # Performs rotation of the sensor data
│   │   ├── transformation.py        # Reconstruct handle forces
│   │   └── dummy_data.py            # Publishes force data for testing
├── path_planning              # Package for path planning
│   ├── config                 # Configuration files
│   ├── launch                 # Launch files
│   ├── src                    # Source code
│   │   ├── planner.py               # Publish steering angle for VAC
│   │   ├── waypoints.py             # Publish next waypoint for guided walk
│   │   └── dummy_data.py            # Publish steering angle for testing
├── recorder                   # Package to record data
│   ├── src                    # Source code
│   │   └── record.py                # Record data from subscribers
├── walking_statistics         # Package to collect walking statistics
│   ├── config                 # Configuration files
│   ├── src                    # Source code
│   │   ├── capture_statistics.py    # Statistics for walking ellipse
│   │   ├── cost_evaluation.py       # Record metrics for cost evaluation
│   │   └── dummy_walker.py          # Publish laser scanner data for testing
└── .gitignore                 # Specifies files to be ignored by Git
```

In addition, each package includes a `package.xml` file, which serves as the package manifest, and a `CMakeLists.txt` file containing build instructions.

Launch scripts

To start all the scripts, you can run the bash script start_all.sh by executing ./start_all.sh in the terminal. This script initiates the following functions:

- Force Sensor
  - `netft_node` for force measurements
  - Rotation and transformation of the force data
  - Calibration scheme to offset the sensor
- Controllers
  - Admittance controller
  - Fall Prevention
  - Guided walk controller
- Path Planners
  - Planner (Guidance)
  - Waypoints (Guided Walk)
- Data Recorder
- Cost evaluation

Services

The various controllers or recorders can now be enabled by interacting with FlexGui on the tablet of the robot or calling the following services from the terminal:

- **Admittance controller** offering walking support.
  - **Start:** `/controller/set/start`
  - **Stop:** `/controller/set/stop`
- **Fall prevention** by adapting the parameters of the admittance controller.
  - **Start:** `/fall_prevention/set/start`
  - **Stop:** `/fall_prevention/set/stop`
- **Guidance** by adapting the parameters of the admittance controller.
  - **Start:** `/controller/guidance/start` & `/planner/set/start`
  - **Start:** `/controller/guidance/stop` & `/planner/set/stop`
- **Guided walk** offering an alternative approach for guidance.
  - **Start:** `/guided_walk/set/start` & `/waypoints/set/start`
  - **Stop:** `/guided_walk/set/stop` & `/waypoints/set/stop`
- **Data recorder and Cost Evaluation.**
  - **Start**: `/recorder/set/start` & `/cost_evaluation/start`
  - **Stop:** `/recorder/set/stop` & `/cost_evaluation/stop`
  - Stopping saves the data that has been recorded from the recorder was started. The data is stored with the filename decided by the rosparam `/filename`.

All these services are of the service type `std_srvs/Empty` and can be executed from the terminal using the command **rosservice call service_name**, where `service_name` is the specific name of the service that is called.

Additionally, since the fall prevention and guidance solutions rely on the admittance controller, they should be halted when the admittance controller is stopped. Moreover, the guided walk cannot operate concurrently with the admittance controller for walking support. Therefore, activating the guided walk requires stopping the admittance controller, and vice versa.

Walking Statistics

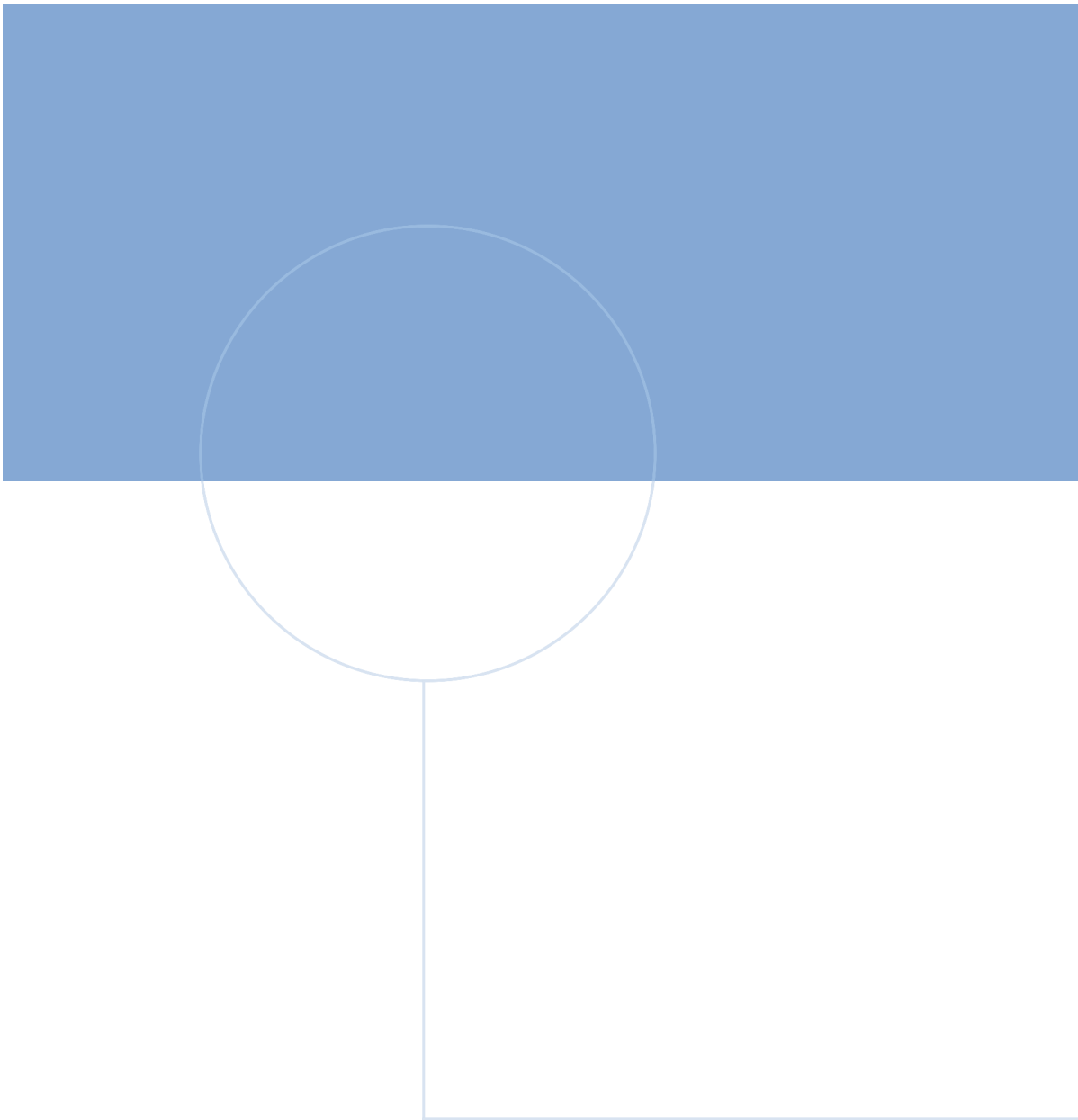
Recording the statistics required for determining the walking ellipse involves the following steps

1. Ensure that walking assistance is activated and the rear laser scanner is publishing data.
2. Launch the `capture_statistics.py` function by executing **rosrun walking_statistics capture_statistics.py** from the terminal.
3. Start to record the user's position behind the Kompaï robot by calling the service `statistics/set/toggle` with **rosservice call statistics/set/toggle**.
4. Walk around with the walker to gather the required data. When finished, call the same service to stop recording. The data is temporarily stored in arrays until reset with the service call **rosservice call statistics/set/reset** or until the node is terminated.
5. The walking statistics are derived from the recorded data when calling the `statistics/get/report` service using the command **rosservice call statistics/get/report**. The statistics are stored with the filename `statistics.yaml` in the config directory of the `walking_statistics` package.
6. After obtaining the walking statistics, the walking ellipse is calculated and represented by its semi-major (a) and semi-minor (b) axes using the command **rosservice call statistics/get/interval**. The walking ellipse is saved in the config directory under the filename `interval.yaml`.

Tips and Tricks

- How to build a catkin package in python: https://dabit-industries.github.io/turtlebot2-tutorials/08b-ROSPY_Building.html
- Make new scripts executable with:

```
chmod +x src/.../.../...py
```



NTNU

Norwegian University of
Science and Technology